EXEMPLAR DE ASSINANTE – VENDA PROIBIDA Para Manaus, Boa Vista, Porto Velho, Rio Branco, Santarém, Altamira, Macapá, Est. de Rondônia (Via Aérea), Cr$ 2.080,00

Cr$ 1.600,00 Ano I – Nº 9 – 1984

A Peregrinação do Dalai Lama

Dissecando: PRINT

Código Morse em seu TK

Curso de Assembly e Basic

Garantia integral

MICROSOFT

MICROSOFT

EDITORIAL

*O artigo **Inteligência Artificial** que publicamos na última edição de **Microhobby**, gerou algumas indagações, realmente merecedoras de discussão. Antes de determinarmos, porém, se uma máquina pode ser inteligente devemos nos perguntar o que é um ser inteligente.*

Atualmente, muitas pessoas sensatas e inteligentes do mundo inteiro estão se perguntando, seriamente, se é válida a matança indiscriminada de baleias e chimpanzés. Os critérios para estas indagações não são de defesa ou respeito à vida em si. Se assim fosse morreríamos todos de fome, pois até um pé de alface é um ser vivo. Na realidade, o princípio ético que nos faz rejeitar esta matança é o do respeito à inteligência, ao ser senciente (pensante).

Alguns primatas, destacadamente o chimpanzé, evidenciam um comportamento e uma aptidão lingüística que os colocam a um nível muito próximo dos seres humanos. Todos os cetáceos (golfinhos, baleias, toninhas e os demais mamíferos aquáticos) exibem códigos de comunicação ainda não bem decifrados, mas que denotam uma inteligência assustadoramente próxima da nossa.

O argumento dos que defendem a matança é linear: ''afinal não passam de animais''. Este é um argumento pura e simplesmente racista, de ''redneck''. É como se, em relação ao holocasto nazista, alguém dissesse ''afinal não passam de ciganos ou judeus''. Ou então, como se um bom argumento anti-abolicionista fosse: ''ora, não passam de negros''.

Em cima da mesa da maioria de nossos leitores, existe uma minúscula semente que, ao germinar, pode produzir algo tão gigantesco que tornará estas polêmicas atuais parecerem, no futuro, frivolidades de adolescentes desocupados (aliás é assim que os ''rednecks'' as classificam).

Esse seu pequeno computador que você programa com tanto prazer, com quem você dialoga apesar de sabê-lo surdo, que você xinga quando não carrega aquele maldito programa daquela maldita fita, passada naquele maldito gravador, que você beija quando descobre que conseguiu eliminar o grilo no programa, é um remoto ancestral maravilhoso, mas assustador.

Num futuro, talvez não tão remoto, a humanidade vai ter que assumir um tipo muito peculiar de paternidade: as máquinas inteligentes e sencientes. O complexo de Frankstein faz com que muitos não aceitem essa possibilidade, mas uma análise fria da curva do progresso tecnológico na área de cibernética (tão impropriamente alcunhada de informática) mostra o aparecimento de máquinas tão ou mais inteligentes que o homem, é praticamente fatal.

Nesse momento vou fazer uma proposta com meus leitores: dentro de 30 anos no máximo, algum órgão do poder Legislativo provavelmente o Congresso americano — vai estar votando uma emenda à Constituição, determinando se é ou não um assassinato o ato de se desligar certas categorias de computadores sem seu prévio consentimento (''seu'' aqui se refere ao computador!).

Só espero duas coisas quando este dia chegar: que estejamos todos vivos para que eu possa cobrar uma cerveja de cada um de vocês, e que possamos olhar para trás e encarar como um pesadelo do passado, aquela época em que nem a inteligência dos próprios sêres humanos era respeitada.

Explicação dos "QUADROS"

Estamos procurando a cada número, melhorar mais a revista, tentando proporcionar a você maiores informações. Esperamos que as inovações introduzidas na revista, estejam agradando.

A partir deste número, todos os programas serão acompanhados por um quadro que fornecerá algumas informações:

1) ***Nível:*** Os programas serão classificados por níveis.

1 computador: para programas até 2 K.

2 computadores: para programas com mais de 2 K em BASIC.

3 computadores: para programas em BASIC e em linguagem de máquina.

Esta classificação de nível de dificuldade dos programas, será feita aqui, por nossa redação.

2) ***Memória ocupada:*** O número de bytes que o programa e as variáveis do sistema ocupam na memória do computador, corresponderá ao valor fornecido em memória ocupada. Este valor é obtido através da instrução: PRINT PEEK 16404 + 256 * PEEK 16405 — 16384

Para programas em que a instrução DIM varia, o valor fornecido será a memória ocupada somente pelo programa que é obtido através da instrução:
PRINT PEEK 16396 + 256 * PEEK 16397 — 16509

3) ***Soma Sintática:*** Estamos publicando novamente o programa Soma Sintática a pedido de muitos leitores que, por terem feito sua assinatura após à sua publicação, escrevem-nos solicitando a listagem deste programa.

Antes de digitar um programa, carregue o programa Soma Sintática e depois coloque-o na RAMTOP. Digite logo em seguida, o programa exatamente como ele é apresentado e verifique se a Soma Sintática confere com o valor fornecido, através da instrução:
PRINT USR (endereço escolhido)

Mas faça a verificação antes de rodar o programa ou fazer alguma alteração, caso contrário, os valores não serão os mesmos.

IMPORTANTE: Para programas que tenham alguma rotina em linguagem de máquina, não será fornecido o valor da Soma Sintática.

4) ***Computador:*** Este último item fornece os computadores em que o programa pode ser digitado. Logicamente, todos os compatíveis não serão incluídos no quadro.

Esperamos que este quadro lhe ajude a conhecer o programa antes de digitá-lo assim como a verificar se você digitou-o corretamente.

SOMA SINTÁTICA

Para criar sua fita com a função ***Soma Sintática,*** você deve executar os seguintes passos:

1. Tecle o programa da ***listagem 1*** para carregar os códigos hexadecimais.

```
  10 REM 0000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
0000000
  20 LET X$=""
  30 LET X=16514
  40 IF X$="" THEN INPUT X$
  50 IF X$(1)="S" THEN STOP
  60 PRINT X$(1 TO 2);"  ";
  70 POKE X,(16*CODE X$+CODE X$(
2)-476)
  80 LET X=X+1
  90 LET X$=X$(3 TO )
 100 GOTO 40
```

2. Execute o programa dando entrada no código hexadecimal da ***listagem 2.***

```
CD  23  0F  21  7D  40  ED  5B
0C  40  FD  21  00  00  CD  A9
40  2A  10  40  ED  5B  14  40
1B  CD  A9  40  FD  E5  FD  21
00  40  CD  2B  0F  C1  C9  E5
A7  ED  52  E1  C8  4E  06  00
FD  09  23  18  F2  21  27  00
ED  5B  04  40  19  FD  21  90
40  FD  75  01  FD  74  02  FD
75  0C  FD  74  0D  FD  21  00
40  21  62  40  01  35  00  ED
B0  ED  4B  04  40  C9
```

Em caso de erro, recomece a segunda etapa.

3. Tecle POKE ***16510,0*** e ***NEW LINE.*** Seu programa aparecerá como a ***listagem 3.***

```
   0 REM LN ??5?RND GOSUB ?£RND
CLEAR 5  LN  RNDE(RND GOSUB ?=RN
D.LN  RND CLEAR FAST CLEAR 5 RND
LN F?AT TAN FAST   GOSUB ? LPRIN
T COS ?  CLEAR  7/ PAUSE 5B  GO
SUB ?.RND; CLEAR 5 RND CLEAR ?
CLEAR ?  CLEAR ?£ CLEAR ?$ CLEAR
 5 RND5£RND P  GOSUB   GOSUB ?.R
NDTAN
  20 LET X$=""
  30 LET X=16514
  40 IF X$="" THEN INPUT X$
  50 IF X$(1)="S" THEN STOP
  60 PRINT X$(1 TO 2);"  ";
  70 POKE X,(16*CODE X$+CODE X$(
2)-476)
  80 LET X=X+1
  90 LET X$=X$(3 TO )
 100 GOTO 40
```

Apague as linhas 20 a 100 e tecle:

```
  10 SAVE "SOMA"
  20 PRINT "SOMA SINTATICA=";USR
 16514
  30 PRINT "CONFERE? S/N"
  40 IF INKEY$="S" THEN GOTO 60
  50 GOTO 40
  60 PRINT "SOMA SINTATICA NO EN
DERECO ";
  70 INPUT RTP
  80 POKE 16389,INT (RTP/256)
  90 POKE 16388,RTP-INT (RTP/256
)*256
 100 PRINT USR 16567
 110 IF INKEY$="" THEN GOTO 110
 120 NEW
```

4. Salve a versão final da ***Soma Sintática*** teclando RUN. Assim ela estará pronta para ser lida corretamente, e deverá mostrar a seguinte mensagem:

```
SOMA SINTATICA=25525
```

Para usar esta mensagem, tecle o endereço para o qual você quer baixar a RAMTOP (por exemplo, ***30000*** se você tem 16 K). A ***Soma Sintática*** será então obtida através da seguinte instrução:

PRINT USR
(endereço que você escolheu)

Portanto antes de carregar o seu programa carregue antes o programa Soma Sintática. Após informar o endereço limite da memória utilizada, tecle qualquer tecla para limpar a área do BASIC.

Expediente

DIRETOR EDITORIAL
Pierluigi Piazzi
JORNALISTA RESPONSÁVEL
Aristides Ribas Fº
EDITOR
Alvaro A. L. Domingues
COORDENAÇÃO EDITORIAL
Ana Lúcia de Alcântara
ASSESSORIA TÉCNICA
Flavio Rossini
José Wilson Tucci
ANÁLISE E REDAÇÃO
Nancy Mitie Ariga
Carlos Eduardo Rocha Salvato
Gustavo Egidio de Almeida
Renato da Silva Oliveira
Roberto Bertini Renzetti
ARTE
Cassiano Roda
Eliana S. Queiroz Yoshihara
Fatima M. Rossini Gouveia
Osmère Sarkis
PRODUÇÃO GRÁFICA
José Carlos Sarkis
COLABORADORES
Bernardo C. Stein
Tanios Hamzo
C. J. Roda
Luiz Tarcísio de Carvalho Jº
GERÊNCIA GERAL E DEPTO. COMERCIAL
Angel D. Zaccaro Conesa
CORRESPONDENTES
Londres — Robert L. Lloyd
New York — Flavio Rossini
Milão — Bruno Origo
PUBLICIDADE
Lúcia Albuquerque
CIRCULAÇÃO E ASSINATURAS
Marcia Regina Dominiqui
MICROHOBBY é editada mensalmente por MICROMEGA PUBLICAÇÕES E MATERIAL DIDÁTICO LTDA., INPI 2992 Livro A. Endereço para correspondência:
Rua Bahia, 1049 — Cx. Postal 54096 — CEP 01296 — São Paulo, SP —
Para solicitar assinaturas (12 números), enviar cheque nominal cruzado à MICROMEGA P. M. D. LTDA., no valor de Cr$ 16.900,00
Tiragem desta Edição: 30.000 exemplares
NÚMERO 9

FOTOLITOS PONTO 813 8793

Impressos nas oficinas da
EDITORA PARMA LTDA.
Fones: 66-3095 - 826-8849
826-7074 - 209-1523
Av. Antônio Bardela, 180
Guarulhos - São Paulo - Brasil
Com filmes fornecidos pelo Editor

Índice

CARTAS DOS LEITORES

Seção de Jogos

Há pouco mais de um mês atrás, assinei a revista ***Microhobby*** e fiquei satisfeito com a sua qualidade.

Entretanto, gostaria de dar uma sugestão: como a maior parte dos leitores, inclusive eu, gosta de jogos e, também, tem memórias com mais de 16 K, eu acho que vocês poderiam publicar uma seção de JOGOS.

Para finalizar, gostaria de ver respondidas duas coisas:

1) É possível acrescentar funções READ e RESTORE no TK?

2) Como digitar um programa em linguagem de máquina?

Luis Henrique Ishida
São Paulo — SP

Caro Luis,

Você deve ter notado que em todas as edições de Microhobby tem sido publicados programas de jogos. Entretanto, não é só de jogos que vive um possuidor de computador. Os jogos são programas e nossa revista dedica-se principalmente a programas. Não vemos necessidade de separá-los numa seção especial.

As instruções READ, DATA e RESTORE podem ser substituídas, sim no TK. Dê uma olhada no Desgrilando da número 7, onde mostramos um caso particular de substituição destas instruções num programa feito para outro computador, a pedido de um de nossos assinantes.

CARTA DO EDITOR

Nesta edição você deve ter notado algumas mudanças nos artigos que compõe a sua revista. Duas novas seções vêm enriquecer nosso material editorial: Dissecando e Vice-versa.

A primeira tem como objetivo mostrar-lhe todos os detalhes de uma determinada instrução, quer você seja um principiante ou já conheça bastante o BASIC e já tenha partido para a linguagem de máquina. A outra tem como função mostrar-lhe como transpor um programa ou uma determinada instrução para outros computadores que não sejam compatíveis como o TK-83 e vice-versa.

Além disso a apresentação dos nossos programas sofreu uma modificação. Agora, além da soma sintática e da memória ocupada pelo programa e variáveis do sistema, mostraremos também o nível de dificuldade apresentado em cada programa, indicando a que faixa de leitores ele foi dirigido (isso não impede que leitores das outras faixas tentem digitá-los).

Com estas duas seções e o novo quadro pretendemos que nenhum de nossos leitores fique sem ter o que ler em nenhum dos próximos números da Microhobby.

O Editor

Quanto a introdução de um programa em linguagem de Máquina, dê uma olhada no número 8 da nossa revista.

O TK e o Direito

Adquiri um computador CP200 e gostaria de saber como utilizá-lo na minha área de trabalho, que é o Direito.

Outra coisa: eu, por enquanto, não conheço nada de BASIC e gostaria de obter as primeiras lições deste curso.

Além disso, gostaria que vocês me fornecessem todos os dados disponíveis para que eu possa entender melhor meu computador.

Vania Suely Porto
Bauru — SP

Prezada Vania,

Sem dúvida, o TK bem como computadores com ele compatíveis (entre eles o CP200) podem ser úteis em qualquer área.

Podemos citar o programa "O TK no Direito Trabalhista e Civil", do Dr. João Carlos Normanha Salles. Outro programa bastante útil para todas as áreas é o "Arquivo", que saiu no mesmo exemplar.

Quanto ao BASIC, sentimos em informar que nem todos os exemplares estão disponíveis. Todavia, o livro BASIC TK, de autoria de Flavio Rossini e Pierluigi Piazzi, contém todas as lições do nosso curso.

CP-300 e TK 85

No colégio em que estudo foi instalada uma sala de computadores e, todos os dias eu e meus colegas reuníamo-nos para tentar fazer alguns programas nos CP-300.

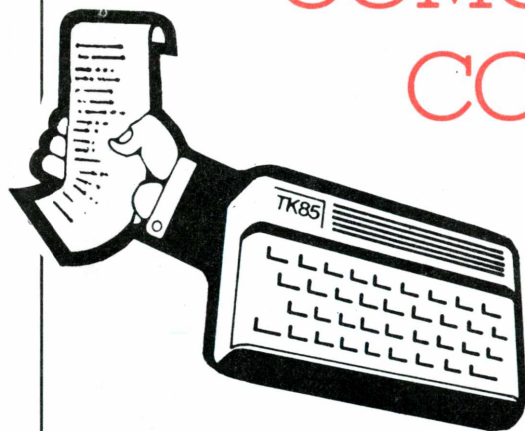
Um deles, certo dia levou uma revista que continha programas especiais para a linha dos TKs, publicada por vocês.

Peguei um destes programas e copiei-o no nosso computador. Nunca fiz nenhum curso de programação, e aquelas eram minhas primeiras experiências com programação. Embora tenha digitado corretamente, o CP-300 acusou erro numa das linhas (talvez por não aceitar a função FAST). A decepção foi enorme. Parecia que aquele computador não aceitava nenhum programa digitado por qualquer um de nós, pois durante mais de uma semana tentamos fazer algo interessante, sem obter resultado.

Conto este caso para poder dizer meu pensamento com maior facilidade: quando se compra um computador e se é quase leigo no assunto, como eu, é necessário uma boa orientação para lidar com ele.

Cláudio Cabaleiro da Costa
São Paulo — SP

Caro Cláudio,

O seu problema é bastante comum, já que apenas alguns computadores são compatíveis com o TK. O CP 300 é compatível com o TRS-80, uma outra linha de computadores, cujo BASIC é essencialmente diferente do BASIC TK. Entretanto, nossa revista possui uma seção, "Os Oitenta", que aborda equipamentos desta natureza. Além disso, colocamos, a partir deste número, a seção Vice-Versa, uma seção voltada a traduções de comandos do BASIC TK para o BASIC dos 80 ou do Apple. O

COMO COLABORAR COM MICROHOBBY

A Revista MICROHOBBY foi criada para servir de intercâmbio entre os leitores que participam do mágico mundo da computação.

A característica realmente inovadora do computador pessoal, está em transformar cada consumidor num criador. Aproveite sua criatividade e envie suas colaborações recebendo remuneração a título de DIREITO AUTORAL.

A maneira ideal de nos enviar o material a ser publicado obedece às seguintes normas:

1. Nunca esqueça de colocar o nome completo, telefone, endereço e número de sua assinatura em **todo** material enviada a nós, sejam listagens de impressora, fitas, envelope, carta ou qualquer outro material.

2. Envie a listagem de programa **datilografada** ou, melhor ainda, tirada na impressora do computador.

3. Coloque sempre uma linha REM com o nome do autor e o título do programa.

4. Envie uma fita com o programa gravado **algumas vezes** (se possível em gravadores diferentes).

5. Na fita, gravar **com microfone** (em viva voz), algumas instruções úteis:
nome completo e **endereço do remetente.**

6. Quando o programa for adaptado e/ou traduzido de outra revista, citar a fonte (autor original, data de publicação, nome da revista e todos os detalhes que houver referente à publicação).

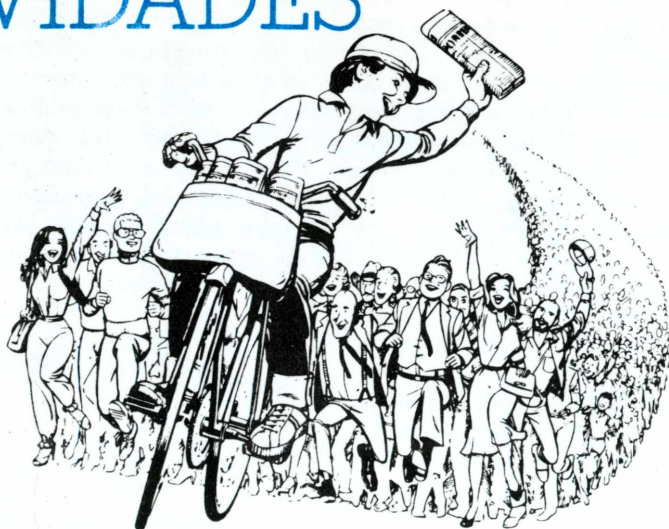
7. Anexar ao material, uma carta autorizando a publicação por parte da revista e assumindo a responsabilidade pela autoria do material e/ou adaptações. Nesta carta, para agilizar a remuneração, podem constar os dados da conta corrente onde daremos o depósito correspondente aos direitos autorais.

8. O material **não utilizado não será devolvido,** ficando a critério da redação a decisão final sobre sua publicação.

9. O material deve ser enviado para:

MICROMEGA PUBLICAÇÕES E MATERIAL DIDÁTICO
SEÇÃO PROGRAMAS DO LEITOR
Cx. Postal 54096
CEP 01296

O

NOVIDADES

Fita cassete que limpa as cabeças de gravação

A BASF está lançando no mercado, uma fita cassete — Hitec High Technology — Head Cleaner ou limpadora de cabeça —. Esta fita pode ser usada em qualquer aparelho, bastando para isso colocá-la no Deck, apertar a tecla PLAY e deixá-la por 1 minuto. Informações com o Serviço de Orientação ao Consumidor da BASF, no telefone: (011) 258-3291.

Um evento de importância na Informática Médica

De 4 a 8 de abril próximo, o Rio de Janeiro, especificamente no Hotel Glória, será centro de um importante evento dentro da área de informática: o I Congresso Brasileiro de Microinformática Médica, com exposição em paralelo: a I Informed.

O objetivo dos organizadores de tal evento é "tentar compensar os 20 anos de atraso em que nos encontramos em relação aos países mais desenvolvidos com referência a este assunto, reunindo os usuários de microcomputadores do Brasil para uma maior compreensão do estágio em que vivemos, para a troca de informações e uma avaliação da importância da Informática Médica".

O evento terá cursos contando com uma programação científica, abordando diversos painéis, conferências, colóquios proferidos por renomados especialistas. A I Informed, realizada em paralelo no Congresso contará com grande número de expositores nacionais, participantes e visitantes mostrando o que está sendo feito no mercado de informática médica.

Periféricos para o TK 82, 83 e 85

Ter um bom computador como o TK 83 é interessante, mas pode não ser suficiente em muitas ocasiões. É útil termos algo mais além do computador, como, por exemplo, uma interface para impressoras paralelas. Este é um dos periféricos lançados este mês pela Microdigital para os computadores da linha TK 83.

Esta interface permite ao usuário o TK 83, 82C e 85 terem seus computadores ligados a uma impressora paralela do tipo Centronic, semelhantemente aos computadores da família TRS-80 ou Apple.

Para aqueles que entendem um pouco mais de eletrônica, a Microdigital está lançando dois módulos interessantes: o programador de EPROM e um Protoborder. O primeiro permite que o usuário programe sua própria EPROM, a partir do TK e o segundo permite que se ligue circuitos eletrônicos ao computador.

Para se ligar mais de um periférico é interessante a placa-mãe (mother border), que permite ao usuário ligar mais de uma expansão ao TK 83, 85, ou 82C.

Um periférico que não pode faltar ao aficcionado em jogos é o joystick, que agora conta com um novo design, mais fácil de ser manipulado.

Para o TK 2000, em breve, estarão no mercado alguns periféricos úteis a seus possuidores:

1) cabo para ligação de impressora
2) interface para disquete
3) joystick

O Micro-Festival

Um outro evento de grande importância, a realizar-se de 21 a 24 de março em São Paulo, no Palácio das Convenções do Anhembi será certamente o Micro-Festival/84. Organizado pela Guazzelli Associados, ele contará com a presença de diversos expositores entre estes: a Microdigital Eletrônica; Verbatim Comercial; a SISCO Sistemas e Computadores; a Multi Soft Informática Ltda.; a Polymax Sistemas e Periféricos; a Itaútec; a Microcraft Microcomputadores; a UNITRON Eletrônica; a Tecnopoli Tecnologia em Poliuretano Indústria e Comércio; a Kristian Eletrônica; a Eletrotela Computadores e Sistemas; a Compucenter Ltda.; a Brascom Computadores; a Scopus Tecnologia; a Victor do Brasil; a Softec Engenharia, Sistemas e Comércio; a Menphis Indústria e Comércio; a Spectrum Equipamentos Eletrônicos; a Exatron Informática; a Sysdata; a Dismac Industrial; a Angra Tecnologia; a Approach Serviços; a Edisa Eletrônica; a Servimec; a Parks Equipamentos Eletrônicos; a Execplan Sistemas de Apoio a Decisão; a Racimec - Racionalização e Mecanização; a VM Consultoria; a Intertec Serviços; a Digiters Indústria e Comércio e Serviços de Eletrônica; a Proceda Serviços Administrativos; a Ritas do Brasil; a Danvic; a Imarés Comércio de Computadores; a Labo Eletrônica; a Computique Comércio e Exportação de Computadores; a Link Comercial; a D.M. Eletrônica, a Rimington; a Compo do Brasil Indústria e Comércio; a SID Sistemas de Informação Distribuída; a Scritta Eletrônica; a BIT Comércio Serviços e Programação Eletrônica; a EMBRATEL; a Elebra Informática; a BMK Processamento de Dados; a CCE Indústria e Comércio de Componentes Eletrônicos, a Milmar Indústria e Comércio e a B.K. Controles Eletrônicos.

Paralelamente à exposição, o Micro-Festival apresentará diversos seminários e cursos, promovidos pela Compucenter. Diversos conferencistas já confirmaram sua presença, entre estes, estão: Claudio Mamana da Sociedade Brasileira de Computação Enrique Costabile, presidente da SUCESU-SP, Sergio Araújo, presidente da ANDEI, Ricardo Saur, da ABICOMP, Helio Azevedo, da SUCESU-Nacional, Miguel Teixeira de Carvalho, da SEI, Ivan da Costa Marque, da Empresa Brasileira de Computação, Frederick Schuchardt, presidente da Schuchardt Software Systems e David Anderson editor da Revista Micro Bits. O

DESGRILANDO

Pentaspeed
Prezados Senhores:

Ao receber o número 4 da revista Microhobby, fui direto ao programa do mês, o Pentaspeed. Digitei o programa e gravei. Quando fui testá-lo, usei o programa Simulador de vôo (um programa longo). Na fase de gravação, deu tudo certo, sendo muito rápido. Entretanto, quando fui carregá-lo, não consegui. Tentei várias vezes com esses e outros programas e continuou não dando certo.

Gostaria de saber o que pode estar havendo. ***Marco Antônio Penna Gonçalves,*** Rio de Janeiro — RJ.

O programa Pentaspeed foi por nós testado, sem nenhum problema. O que pode estar acontecendo é que, por ser o Pentaspeed um programa que vai para a RAMTOP ele não pode coexistir com programas em que, pelo menos uma parte, vá para a RAMTOP. Para resolver este problema, serão necessários alguns artifícios. Nossa equipe está trabalhando no momento nisso, numa das próximas edições, mostraremos o resultado.

Aguarde.

Prezados Senhores:

Possuo um TK-85 com 16 K de memória e gostaria de saber qual o máximo endereço para que ela possa dar um POKE? Idem para 48 K e 64 K?

Estou realmente mais interessado em Linguagem de Máquina que no BASIC do TK. É possível fazer esta inversão (BASIC para Linguagem de Máquina?) sem grandes "dores de cabeça"? Gostaria de fazer isso o mais rapidamente possível. Existem cursos de Linguagem de Máquina por correspondência? Quais? ***Odilon dos Santos Braga,*** Acesita — Minas Gerais.

Caro Odilon,

Você pode fazer um POKE em todos os endereços da RAM do micro. Os endereços máximos são:

Para 16 K 32767
Para 48 K . . . . . . 65535 (no TK 85)
Para 64 K 65535

Quanto a Linguagem de Máquina ela não requer "inversões". Seu computador pode funcionar perfeitamente tanto em uma quanto em outra. Ela é acessada pelo BASIC através das funções PEEK, POKE e USR.

Mais detalhes são mostrados à cada edição de Microhobby. Quanto aos cursos não conhecemos nenhum que ministre aulas por correspondência sobre Linguagem de Máquina específico para o TK ou compatíveis.

Entretanto, você poderá aprender muito a respeito, nas páginas da Microhobby e no livro de Flavio Rossini, Curso Linguagem de Máquina para o TK, editado aqui, pela Micromega.

High speed e Penta speed
Prezados Senhores:

Sou assinante da Microhobby e estou satisfeita com minha assinatura, pois além de ser uma ótima revista, estou recebendo-a assiduamente.

O motivo desta carta é pedir um esclarecimento sobre o Pentaspeed, pois não é com todos os programas que ele funciona, como, por exemplo, "Mazogs" e outros, pois ele simplesmente os rejeita na hora de rodar o programa.

Eu gostaria de saber:

1) Tem solução para isso?

2) Se publicaram o Pentaspeed, por que não o HI-speed? ***Beatriz H. A. De Carvalho*** — Campinas — SP.

Como dissemos na resposta à carta do Marco Antonio, existem alguns macetes que podem permitir que os programas que têm uma de suas partes protegidas pela RAMTOP possam também rodar com o Pentaspeed. Quanto ao HI-speed, não está nos nossos projetos atuais publicar um programa desta natureza.

Agradecemos as suas sugestões e elogios. ○

PRINT

Carlos E. R. Salvato

Está se iniciando neste número uma nova seção: Dissecando.

Dissecar significa, de acordo com Mestre Aurélio, "analisar minuciosamente". E é isso que iremos fazer, abordando a cada mês uma instrução, função ou comando diferentes.

Dividiremos esta seção em três partes:

a) ***Iniciantes:*** destinada aos que estão principiando o uso do computador, dando desde como digitá-lo até sua função num programa.

b) ***Minúcias:*** nesta parte, descreveremos detalhes interessantes, dicas do uso de cada instrução, comando ou função.

c) ***Linguagem de máquina:*** destinada aos que já conhecem bem o BASIC TK e estão iniciando o aprendizado em linguagem de máquina.

Assim teremos, em todos os níveis, informações úteis não só para o iniciante como também para todos aqueles que desejam saber um pouco mais sobre o BASIC e a Linguagem de Máquina do TK.

Nota: usamos para testar um computador TK 83. As funções obtidas com PEEK e POKE e as rotinas em linguagem de máquina podem funcionar na maioria dos computadores compatíveis com ele (TK-82, TK-85, CP200 e NE-Z8000). Talvez faça exceção o Ringo, o AS 1000 e o Apply 300. Um rápido exame dos respectivos manuais poderá dar-lhe alguma orientação de como proceder.

Sem dúvida o PRINT é uma das funções mais usadas nos programas em BASIC. Ele pode ser usado em vários casos e nós, neste artigo, tentaremos mostrar como ele é usado em todos estes casos.

INICIANTES

Dentre outras funções, o PRINT pode ser usado como um comando direto, ou seja, não fará parte de um programa.

Execute a seguinte linha:

```
PRINT 3+1
```

Logo após você ter pressionado NEW LINE, o resultado da conta aparecerá no canto superior esquerdo do seu vídeo. Neste caso, o PRINT fez o computador agir como calculadora.

Se você executar a linha que segue, qual será o resultado?

```
PRINT "3+1"
```

O micro não deu o resultado da conta porque foi ordenado a ele que imprimisse no vídeo o que estivesse entre aspas, no caso, 3 + 1.

Para fazer o TK imprimir a conta e o resultado, você deve fazer o seguinte:

```
PRINT "3+1";3+1
```

O micro imprimiu o que estava entre aspas e por causa do ponto e vírgula, ele imprimiu logo ao lado da conta, o resultado de 3 + 1.

Existem várias formas de se posicionar uma palavra ou número no vídeo, ou seja, seus programas não precisam ir escrevendo as mensagens ou contas começando na parte superior do vídeo. Eles podem começar de qualquer ponto que você desejar.

O VÍDEO

O vídeo do TK é dividido em 32 colunas e 24 linhas, sendo que as linhas 22 e 23 são reservadas para edição e normalmente não podem ser usadas no programa (Mais adiante veremos como usá-las).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	1
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	2
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	3
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	4
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	5
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	6
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	7

Figura 1

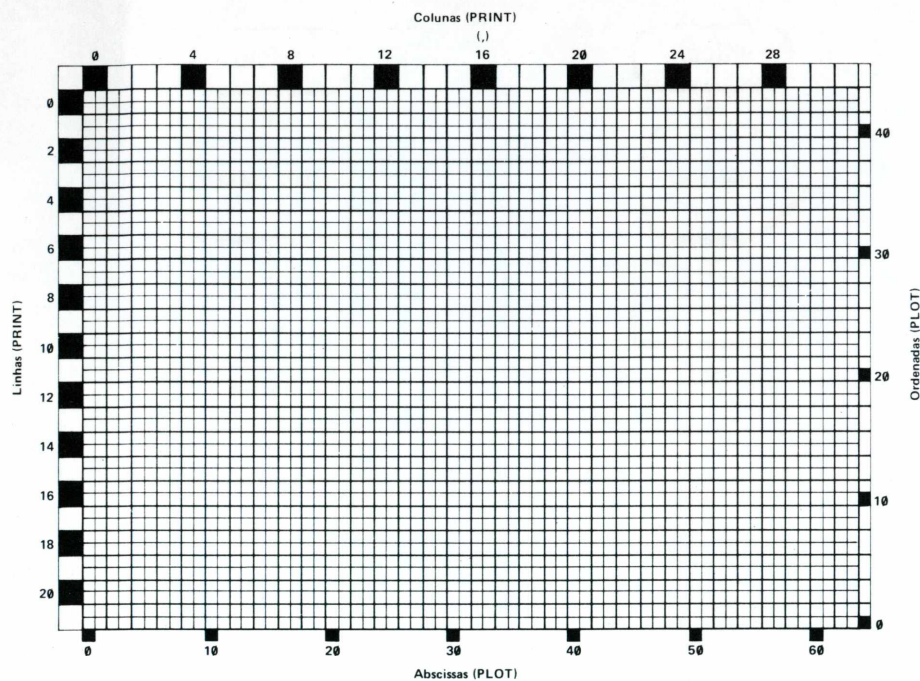



Tabela de formatação do vídeo

PRINT;

Ele serve para escrever uma coisa logo após a outra como já foi citado no exemplo anterior.

PRINT,

O PRINT, divide o vídeo em duas partes. Uma que começa na posição 0 e vai até o 15, e outra que começa na posição 16 e vai até a 31.

Execute a seguinte linha:

```
PRINT 2,2
```

O TK escreverá o primeiro 2 na posição (Ø,Ø) e o segundo 2 na posição (Ø,16).

Se você, em vez de digitar somente uma vírgula digitasse duas, iriam aparecer um 2 em baixo do outro, porque o micro imprimiu o primeiro 2 e depois recebeu a ordem de imprimir duas metades a frente do outro número 2.

PRINT TAB

O PRINT TAB serve como tabulador. Ao invés dele começar a imprimir na posição (Ø,Ø), o micro começará a imprimir na posição (Ø,X) onde X é igual a um número determinado por você, de Ø a 31.

Experimente executar esta linha:

```
PRINT TAB 8;"MICROHOBBY"
```

O TK colocará na posição (0,8), a palavra MICROHOBBY. Se você executar um PRINT TAB X, onde X é um número maior que 31 e menor 256, como por exemplo:

```
PRINT TAB 235;"MICROHOBBY"
```

O computador não acusará erro, porque ele divide os números de Ø até 255 em blocos de 32 como está representado na figura 1.

Você deve estar fazendo a seguinte pergunta: "e se em um programa estiver escrito o seguinte: ".

```
PRINT TAB 181;"MICROHOBBY"
```

"Em que posição será escrita a palavra MICROHOBBY?" Para isto, eu desenvolvi a fórmula que segue:

$$X = N - 32 * L$$

onde: N = o número após o TAB
L = o número da linha onde está N. (ver Figura 1).

Usando esta fórmula, você descobrirá X (que é a posição ocupada pela palavra).

PRINT AT

Este tipo de PRINT é usado da seguinte forma:

PRINT AT Y,X;"MICROHOBBY"

onde Y é o número da linha e X é o número da coluna que você quer posicionar a palavra MICROHOBBY. Serão dadas dicas mais adiante.

"O que devo fazer para aparecer, no início do vídeo, uma palavra entre aspas?".

Se você escrever:

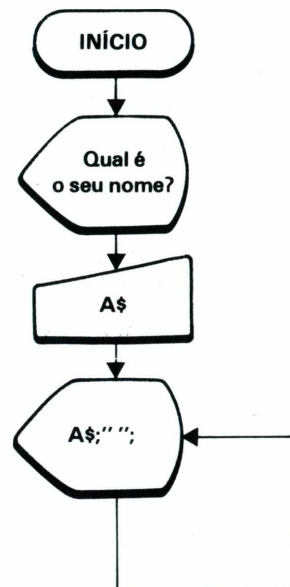
```
PRINT ""MICROHOBBY""
```

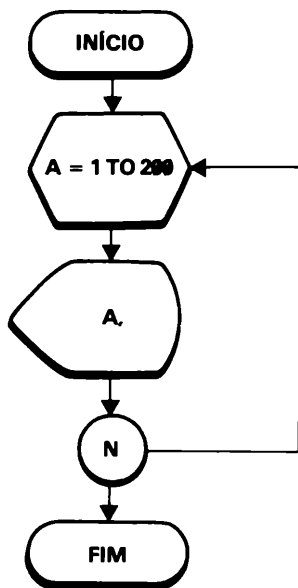
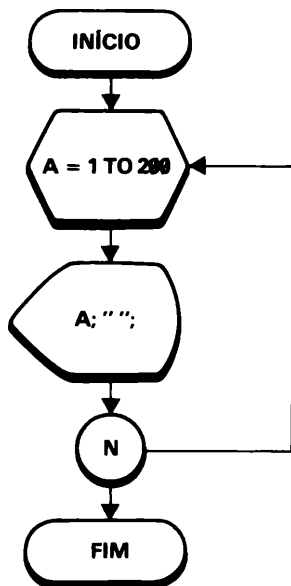
o TK irá acusar erro. Você deve escrever:

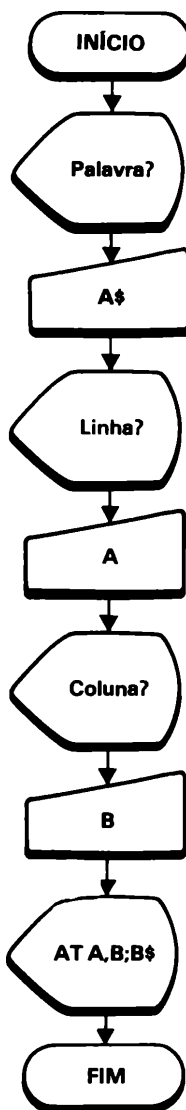
PRINT no programa

Agora nós iremos apresentar alguns programas bem simples para ilustrar o que foi dito anteriormente.

Primeiro iniciaremos com um programa que todos, logo que compram um microcomputador, querem fazer.

```
10 PRINT "QUAL E* O SEU NOME ?"
20 INPUT A$
30 CLS
40 PRINT A$;" ";
50 GOTO 40
```

Para você sentir bem a diferença entre o PRINT; e o PRINT, , digite os dois programas que seguem.

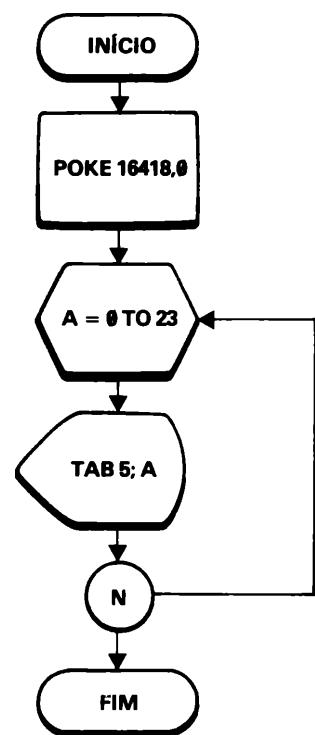
```
10 FOR A=1 TO 200
20 PRINT A;" ";
30 NEXT A
40 STOP
```

```
10 FOR A=1 TO 200
20 PRINT A,
30 NEXT A
40 STOP
```

O programa que segue serve para ilustrar bem o PRINT AT ;

```
  10 PRINT "QUAL A PALAVRA QUE V
OCE QUER     ESCREVER ?"
  20 INPUT A$
  30 PRINT "EM QUE LINHA VOCE QU
ER IMPRIMIR A PALAVRA ?"
  40 INPUT A
  50 PRINT "E EM QUE COLUNA ?"
  60 INPUT B
  70 CLS
  80 PRINT AT A,B;A$
  90 STOP
```

MINÚCIAS

Nesta parte do artigo nós iremos dar algumas dicas sobre o uso do PRINT.

A primeira delas será como usar as linhas de edição. Para fazer isto, basta acrescentar no início de seu programa uma linha que é o POKE 16418,0. Neste endereço está armazenado o valor das linhas de edição que normalmente é 2 e quando o TK executa a linha 16418,0, você está zerando as linhas de edição podendo usar todo o vídeo.

Digite o seguinte programa:

```
10 POKE 16418,0
20 FOR A=0 TO 23
30 PRINT TAB 5;A
40 NEXT A
50 STOP
```

Mas tome cuidado! Às vezes este POKE pode fazer o seu programa sumir, porque você zera o número de linhas de edição e não poderá fazer um INPUT. Tente rodar o programa seguinte:

```
10 POKE 16418,0
20 INPUT A
```

Logo após ter digitado RUN, o vídeo ficará parado e se você digitar qualquer número ou letra, tudo sumirá.

Para resolver isto, antes de fazer um INPUT digite uma linha contendo POKE 16418,2.

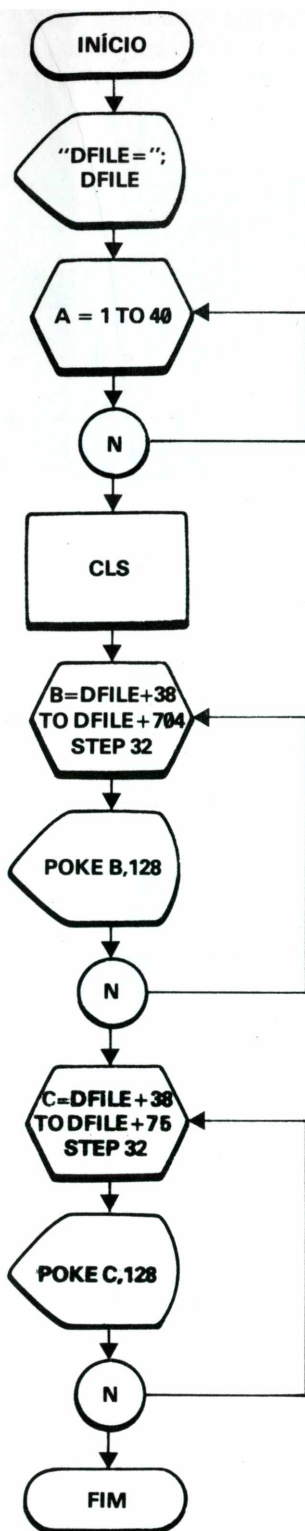
Agora vamos ver como usar as variáveis do sistema nos nossos programas.

1º) DFILE — Arquivo de vídeo

Esta variável está contida em dois endereços que são: 16396 e 16397.

Ela nos indica o endereço onde começa o arquivo de vídeo, porque ele depende do tamanho do programa.

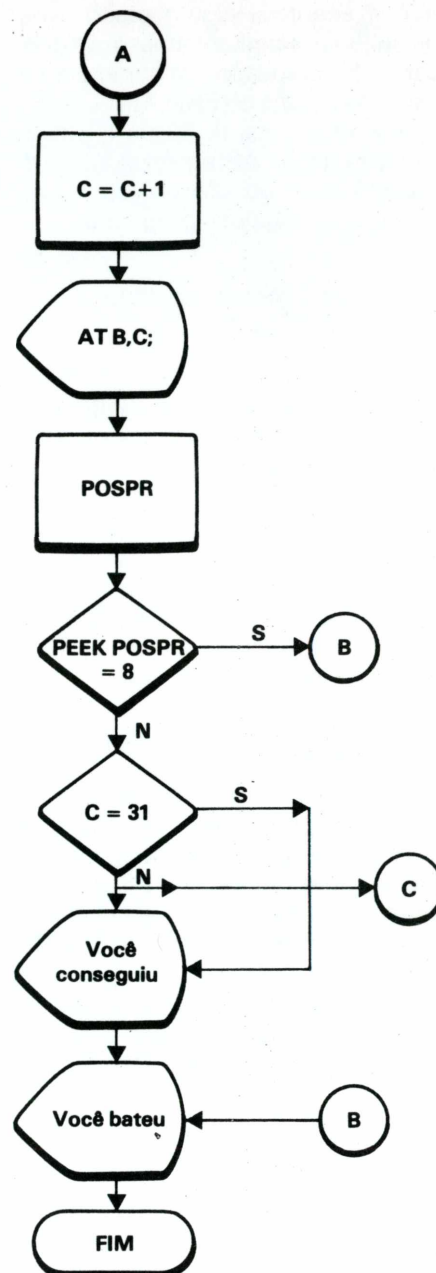
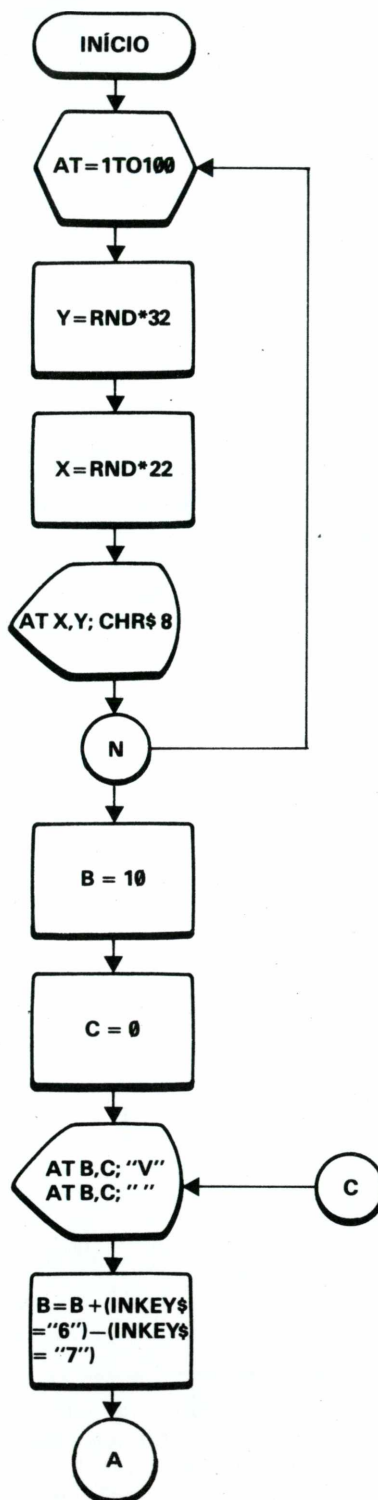
Experimente digitar o seguinte:

```
  10 LET DFILE=PEEK 16396+256*PE
EK 16397
  20 PRINT "O DFILE COMECA NO EN
DERECO ";DFILE
  30 FOR A=1 TO 40
  40 NEXT A
  50 CLS
  60 FOR B=DFILE+58 TO DFILE+704
 STEP 32
  70 POKE B,128
  80 NEXT B
  90 FOR C=DFILE+38 TO DFILE+750
 STEP 34
 100 POKE C,128
 110 NEXT C
 120 STOP
```

A linha 10 define o valor do DFILE. Da linha 60 até a 110 são dois LOOPS que farão imprimir um enorme X no vídeo. Isto é feito "pokeando" o caractere 128 em alguns endereços do DFILE.

2º) POSPR — endereço da posição do PRINT

Esta variável está armazenada nos endereços 16398 e 16399.

Ele testa qualquer ponto e é muito usado em jogos como por exemplo o que segue:

```
  10 FAST
  20 FOR A=1 TO 100
  30 LET Y=INT (RND*22)
  40 LET X=INT (RND*32)
  50 PRINT AT Y,X;CHR$ 8
  60 NEXT A
  70 SLOW
  80 LET B=10
  90 LET C=0
 100 PRINT AT B,C;"■";AT B,C;" "
 110 LET B=B+(INKEY$="6")-(INKEY
$="7")
 120 LET C=C+1
 130 PRINT AT B,C;
 140 LET POSPR=PEEK 16398+256*PE
EK 16399
 150 IF PEEK POSPR=8 THEN GOTO 2
00
 160 IF C=31 THEN GOTO 180
 170 GOTO 100
 180 PRINT AT 10,0;"VOCE CONSEGU
IU NAO BATER EM NADA"
 190 GOTO 210
 200 PRINT AT 10,0;"VOCE BATEU N
O OBSTACULO"
 210 STOP
```


Neste jogo você tem que desviar dos obstáculos, usando as teclas 6 e 7.

Mas o que nos interessa mesmo são as linhas 130, 140 e 150.

A linha 130 posiciona um cursor que não é visível na posição (C,B); a linha 140 atribui um valor à variável POSPR e a linha 150 testa que caractere existe nesta posição e compara com o caractere 8 que simboliza o obstáculo. Se o que houver na posição (C,B) for o caractere 8, é feito um salto para a linha que imprime a mensagem "VOCE BATEU", caso contrário, o programa continua normalmente.

3º) COLPR — Número de coluna para a posição PRINT

Esta variável está armazenada somente em um endereço, o 16441.

Ela é muito útil quando você quizer que aconteça algum efeito especial ou uma simples limpeza de tela, quando fôr colocado um caractere em uma certa coluna do PRINT.

Ela pode variar de 32 até 1, ou seja a posição 0 do TAB equivale ao 32 da variável e assim por diante.

Rode o seguinte programa:

```
10 FOR A=1 TO 352
20 PRINT ".";
40 NEXT A
50 STOP
```

O TK encheu até a metade do vídeo com pontos. Agora inclua a seguinte linha. (Fig. 21)

```
30 IF PEEK 16441=1 THEN PRINT
,,
```

Quando o TK chega na linha 30 ele testa se a variável COLPR é igual a 1. Se for ele pula uma linha se não, continua o programa.

4º) LINPR — Número da linha para a posição PRINT

Esta variável também está armazenada em apenas um endereço, o 164 42. A função dela é quase igual a da COLPR, mas agora ela servirá para identificar a linha e não a coluna do PRINT. Ela é muito usada para evitar o erro 5.

```
10 PRINT TAB 8;"MICROHOBBY"
20 IF PEEK 16442=2 THEN CLS
30 GOTO 10
```

Aqui também as posições são invertidas do PRINT AT. Quando a linha no PRINT AT é igual a 0 nesta variável a linha é igual a 23.

LINGUAGEM DE MÁQUINA

Esta parte do artigo será dedicada àqueles que já entendem bem o BASIC-TK e já tiveram alguma introdução a Linguagem de Máquina.

Para carregar o programa Linguagem de Máquina é necessário usar um programa em BASIC por nós denominado de Monitor Assembly. O Monitor é um programa simples que "pokeia" na linha REM os caracteres equivalentes aos mnemônicos da Linguagem de Máquina. A primeira linha do Monitor Assembly sempre deverá ser a linha REM com N zeros, sendo que N varia de programa para programa. N sempre será especificado no início do programa em Linguagem de Máquina. Os valores dos endereços inicial e final também são especificados no início do programa.

Após você ter entrado com os endereços digite os números hexadecimais logo abaixo da palavra Código. Os códigos deverão ser digitados um a um, ou seja após digitar um código digite NEW LINE.

Quando aparecer no canto inferior esquerdo o código de reportagem 9/160 apague todas as linhas após a linha 20 (a linha 20 também deve ser apagada), não use NEW.

Para rodar um programa em Linguagem de Máquina digite:

```
RAND USR 16514
```

```
 10 REM  N ZEROS
 20 PRINT "ENDERECO INICIAL ?"
 30 INPUT A
 40 PRINT A
 50 PRINT "ENDERECO FINAL ?"
 60 INPUT B
 70 PRINT B
 80 FOR C=A TO B
 90 SCROLL
100 PRINT C,
110 INPUT A$
120 PRINT A$
130 LET D=16*CODE A$+CODE A$(2)
-476
140 POKE C,D
150 NEXT C
160 STOP
```

Agora nós iremos fazer algumas comparações entre os programas em BASIC e os programas em Linguagem de Máquina.

Os dois programas seguintes preencherão o vídeo inteiro com a palavra MICROHOBBY em vídeo inverso.

a) Versão BASIC

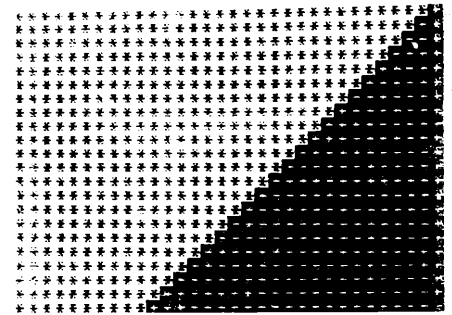
```
10 PRINT "MICROHOBBY";
20 GOTO 10
```

b) Versão ASSEMBLY

```
N=40
ENDERECO INICIAL=16514
ENDERECO FINAL=16544

ENDERECO   CODIGO     MNEMONICOS
16514      18 0B      JR      +0B
16516      B2         M
16517      AE         I
16518      A6         C
16519      B7         R
16520      B4         O
16521      AD         H
16522      B4         O
16523      A7         B
16524      A7         B
16525      BE         Y
16526      80
16527      C5         PUSH    BC
16528      E5         PUSH    HL
16529      D5         PUSH    DE
16530      11 84 40   LD DE,4084
16533      01 0B 00   LD BC,000B
16536      CD 6B 0B   CALL    0B6B
16539      D1         POP     DE
16540      E1         POP     HL
16541      C1         POP     BC
16542      CD 8F 40   CALL    408F
```

A versão BASIC é muito mais fácil de ser digitado porém a velocidade de processamento é inferior a da versão Assembly. A versão em Assembly por sua vez tem sua digitação meio crítica, qualquer número errado poderá causar efeitos estranhíssimos.

Agora iremos apresentar dois programas que terão como resultado final a figura:

a) Versão BASIC

```
 10 FOR A=1 TO 704
 20 PRINT "*";
 30 NEXT A
 40 LET B=10
 50 FOR C=21 TO 0 STEP -1
 60 FOR D=B TO 31
 70 PRINT AT C,D;"■"
 80 NEXT D
 90 LET B=B+1
100 NEXT C
110 STOP
```

b) Versão ASSEMBLY

```
N=33
ENDERECO INICIAL=16514
ENDERECO FINAL=16546

ENDERECO  CODIGOS   MNEMONICOS
16514     2A 0C 40  LD     HL,(400C)
16517     23        INC    HL
16518     06 16     LD     B,16
16520     0E 1F     LD     C,1F
16522     3E 17     LD     A,17
16524     51        LD     D,C
16525     1E 20     LD     E,20
16527     77        LD     (HL),A
16528     23        INC    HL
16529     1D        DEC    E
16530     15        DEC    D
16531     C2 8F 40  JP     NZ,408F
16534     3E 97     LD     A,97
16536     77        LD     (HL),A
16537     23        INC    HL
16538     1D        DEC    E
16539     C2 98 40  JP     NZ,4098
16542     23        INC    HL
16543     0D        DEC    C
16544     10 E8     DJNZ   408A
16546     C9        RET
```

Outra vez a versão Assembly se destaca por ter uma execução mais rápida. O problema do programa BASIC é que ele encheu o vídeo inteiro com asteriscos normais e depois em algumas posições ele coloca asteriscos em vídeo inverso.

Agora digite a linha seguinte:

```
1 REM LN ■■LN ■RND
2 RAND USR 16514
```

Logo após ter digitado isto o que aconteceu? Você sabe explicar? Nos próximos números nós dissecaremos outros comandos. Até lá. ○

MENSAGEM MORSE ESCRITA E SONORA

Código Morse
Memória Ocupada = 3072 bytes
Nível =
Computadores = TK-85 e compatíveis

Rodnei Novaes

Este programa pede na tela para você entrar com o texto e a partir daí, este é codificado em MORSE, isto é, traço e ponto obedecendo a codificação internacional. (Fig. 1)

```
ENTRE COM O TEXTO

MICROHOBBY
-- .. -.-. .-. --- .... --- -...
 -... -.--

AUMENTE O VOLUME DA TV E NEWLINE
```

Logo a seguir aparece no vídeo: "AUMENTE O VOLUME DA TV E NEW LINE".

Com isso, cada ponto é sonorizado com um sinal de longa duração. Com os cabos ligados deixando o gravador em RECORD, você grava os dados (sinais) e com um fone de ouvido ligado na saída do monitor a reprodução é ainda melhor que a mostrada na TV.

Para digitar o programa, proceda da seguinte forma:

— Digite primeiro o monitor assembler listado na figura 2.

```
   1 REM 000000000000000000000000
0000000000
  10 LET X=16514
  20 PRINT " ";
  30 PRINT X;"> ";
  40 INPUT A$
  50 IF A$="S" THEN STOP
  60 POKE X,VAL A$
  70 PRINT VAL A$;
  80 LET X=X+1
  90 GOTO 20
```

— A linha 1 REM contém 33 zeros.

— Após digitado o programa, dê o comando RUN e NEW LINE. Entre agora com os códigos em decimal, listados na figura 3.

```
6      16514
40     16515
30     16516
64     16517
167    16518
187    16519
40     16520
12     16521
219    16522
254    16523
205    16524
158    16525
64     16526
211    16527
255    16528
205    16529
158    16530
64     16531
24     16532
5      16533
14     16534
255    16535
13     16536
32     16537
253    16538
16     16539
231    16540
201    16541
123    16542
61     16543
32     16544
253    16545
201    16546
```

— Tendo digitado todos os códigos, digite S e NEW LINE.

— Delete agora todas as linhas, menos a de número 1.

— Digite agora a segunda parte, isto é, o programa propriamente dito. (Fig. 4).

```
   1 REM ▪"C2RND▒C£<= RETURN LN
▒RNDPEEK COPY LN ▒RND/▪: COPY $4
 CLEAR ( SCROLL TAN ?X4 CLEAR TA
N 00

   2 REM RODNEI NOVAIS
  90 DIM B$(40,8)
  97 LET B$(1)="-.-.-."
  98 LET B$(2)=".-.-.-"
  99 LET B$(3)=".. .. .."
 100 LET B$(4)="-----"
 101 LET B$(5)=".----"
 102 LET B$(6)="..---"
 103 LET B$(7)="...--"
 104 LET B$(8)="....-"
 105 LET B$(9)="....."
 106 LET B$(10)="-...."
 107 LET B$(11)="--..."
 108 LET B$(12)="---.."
 109 LET B$(13)="----."
 110 LET B$(14)=".-"
 111 LET B$(15)="-..."
 112 LET B$(16)="-.-."
 113 LET B$(17)="-.."
 114 LET B$(18)="."
 115 LET B$(19)="..-."
 116 LET B$(20)="--."
 117 LET B$(21)="...."
 118 LET B$(22)=".."
 119 LET B$(23)=".---"
 120 LET B$(24)="-.-"
 121 LET B$(25)=".-.."
 122 LET B$(26)="--"
 123 LET B$(27)="-."
 124 LET B$(28)="---"
 125 LET B$(29)=".--."
 126 LET B$(30)="--.-"
 127 LET B$(31)=".-."
 128 LET B$(32)="..."
 129 LET B$(33)="-"
 130 LET B$(34)="..-"
 131 LET B$(35)="...-"
 132 LET B$(36)=".--"
 133 LET B$(37)="-..-"
 134 LET B$(38)="-.--"
 135 LET B$(39)="--.."
 136 LET B$(40)="..--.."
 195 CLS
 200 PRINT "ENTRE COM O TEXTO"
 205 PRINT
 206 INPUT A$
 208 PRINT A$
 210 LET D$=""
 240 FOR I=1 TO LEN A$
 241 IF A$(I)="." THEN GOTO 500
 242 IF A$(I)=" " THEN GOTO 350
 243 IF A$(I)="?" THEN GOTO 400
 245 LET C$=""
 250 FOR J=1 TO LEN B$(CODE A$(I
)-24)
 260 IF B$(CODE A$(I)-24,J)=" "
THEN GOTO 290
 270 LET C$=C$+B$(CODE A$(I)-24,
J)
 280 NEXT J
 290 PRINT C$;" ";
 295 LET D$=D$+C$
 300 NEXT I
 302 PRINT AT 21,0;"AUMENTE O VO
LUME DA TV E NEWLINE"
 305 IF INKEY$="" THEN GOTO 305
 310 GOTO 790
 350 LET A$(I)="/"
 355 LET C$=A$(I)
 360 GOTO 290
 400 LET C$=B$(40)
 410 GOTO 290
 500 LET C$=B$(3)
 510 GOTO 290
 600 SAVE "MORSE"
 610 RUN
 790 FAST
 800 FOR I=1 TO LEN D$
 810 IF D$(I)="." THEN GOSUB 900
 820 IF D$(I)="-" THEN GOSUB 100
0
 830 IF D$(I)="/" THEN GOSUB 110
0
 835 IF D$(I)=" " THEN POKE 1651
7,0
 840 RAND USR 16514
 842 FOR J=1 TO 10
 843 NEXT J
 850 NEXT I
 860 GOTO 1200
 900 POKE 16515,40
 910 POKE 16517,64
 920 RETURN
1000 POKE 16515,255
1010 POKE 16517,64
1020 RETURN
1100 POKE 16517,0
1110 FOR J=1 TO 10
1120 NEXT J
1130 RETURN
1200 SLOW
1210 PRINT AT 21,0;" QUER REPETI
R A MENSAGEM ? (S/N)"
1220 INPUT S$
1230 IF S$="S" THEN GOTO 790
1240 IF S$="N" THEN GOTO 195
1250 GOTO 1220
```

— GRAVE o programa e dê o comando RUN (new line).

VICE-VERSA

C. J. Roda

Esta seção pretende interligar os software dos micros pessoais mais conhecidos no nosso país, na medida do possível. Por que? Porque muitas máquinas são extremamente semelhantes. Outras tem uma concepção de projeto que visa usuários diferentes. Estas diferenças também podem existir no tipo de linguagem utilizada (Basic, Pascal, Fortran, etc.). Vamos comentá-las na Vice Versa, para que o usuário de uma determinada máquina possa beneficiar-se do material publicado para as outras. Afinal não há razão para que um usuário do TK fique sem entender a listagem de um programa criado para outro computador, principalmente se esta listagem estiver em Basic!

A compatibilidade

Quando pela primeira vez alguém vai a uma loja de micros sai de lá espantado com a quantidade enorme de marcas de máquinas em exposição. A impressão que se tem é que deve existir ali uma centena de tipos de computador, cada um com um modo particular de programação. A coisa não é bem assim. Na realidade com pequenas diferenças, existem praticamente só três ou quatro famílias de micros . . . que embora de marcas diferentes, rodam praticamente os mesmos programas. A este conjunto de computadores que podem rodar praticamente o mesmo software e as vezes até intercambiar periféricos dá-se o nome de ***compatíveis.*** Agora antes de mais nada, dê uma olhada na ***tabela de compatibilidade.***

Tabela de compatibilidade
TK 82, TK 82C, TK 83, CP 200, NEZ 8000, TK 85, AS 1000, APPLY 300 RINGO
APPLE, UNITRON, ELPPA, APII, APPLETRONICS, SPECTRUM, MAXXI, D 81000, EXATO MICROENGENHO I, II
TRS 80 (MOD I), D8000, D8001, D8002, DGT 100, DGT 101.
TRS 80 (MOD III), JRSYSDATA, CP300, CP500, NAJA IBM, EGO, NEXUS, CP 2001 SCOPUS.
TK2000, COMPATÍVEL C/ APPLESOFT

Os micros dentro da mesma célula são o que definimos por compatíveis. É bom notar que a tabela não está completa (surge uma nova marca de micros por dia . . .) e também é bom que fique claro não me canso de repetir) ***a compatibilidade perfeita não existe!***

Droga! Este programa não passa no meu micro!

Esta expressão deve ser a mais ouvida dentro das livrarias técnicas e bancas que vendem revistas importadas sobre computação. A verdade é que, não importa qual micro você possui, sempre existirão mais programas para os outros micros . . . (a maçã do vizinho sempre é mais vermelha) ***"Mas caramba! Este negócio parece que está escrito em Basic, mas não parece o meu Basic . . . E o que são estes POKES aqui? Eu tenho POKE & PEEK no meu computador mas . . . E estes caracteres gráficos, o que são? . . . E esta linha REM esquisita, que será? . . . Uai, aqui diz que tem uma rotina em linguagem de máquina, mas eu não vejo linha REM . . . Como será que eles fazem? . . . DROGA!"***

Estes pensamentos e outros, talvez até impublicáveis, são resultado de

uma incompatibilidade de gênios entre o seu computador e um outro qualquer . . . Mas porque não "adaptar" de uma máquina para outra? Mas como? Entendendo o que uma outra máquina faz. Você mesmo pode, com um pouco de imaginação e algum trabalho, ***"fazer a sua máquina fazer".***

Vice-Versa

Para iniciar a seção vamos dar uma olhada na forma com que o TK e o TRS-80 armazenam suas linhas de programa. O objetivo é sugerir, ao possuidor do TRS-80 e do TK, algumas idéias de como criar um renumerador e, possivelmente, um programa que crie outros programas. Numa segunda parte do artigo (próximo número) mostraremos as tabelas de caracteres alfanuméricos, gráficos e o código ASCII, utilizados pela maior parte dos computadores, mas ***não*** pelo TK.

Vamos a la digitacíon?

Se você tem TRS-80 modelo I* ou compatível ou TK ou compatível, digite o programa da figura 1.

O PROGRAMA

```
  10 REM 0000*
  20 REM VICE-VERSA - MICROHOBBY
  30 PRINT "TRS-80 (T) OU TK (S)
"
  40 INPUT A$
  50 IF A$="T" THEN LET CL=17129
  60 IF A$="T" THEN LET FL=0
  70 IF A$="S" THEN LET CL=16509
  80 IF A$="S" THEN LET FL=118
  90 IF A$<>"S" AND A$<>"T" THEN
 GOTO 40
 100 PRINT "COMECO DA LINHA ";CL
 110 PRINT "CODIGO DE FIM DE LIN
HA ";FL
 120 PRINT CL;" ";PEEK (CL);" ";
CHR$ (PEEK (CL))
 130 IF PEEK (CL)=FL THEN PRINT
 140 LET CL=CL+1
 150 GOTO 120
```

Figura 1

Não se assuste. Não importa se você acha que esta é uma listagem de TK. Ela foi escrita para rodar, sem alterações, também no TRS-80. Atenção rápidos no gatilho, digo, no teclado! a linha REM precisa ser digitada, neste caso, porque é dela que iremos falar.

Rode o programa. O programa irá perguntar em que máquina está, TRS-80 ou TK. Assim que você responder, ele irá mostrar na tela, numa mesma linha, um endereço de memória (I), um byte (PEEK (I)) e um caractere ou comando (CHR$ (PEEK (I)). (Atenção possuidores de 80s! Se for preciso vocês terão que parar a impressão dos dados, que será contínua, usando BREAK, assim que a tela estiver cheia. O TK para automaticamente. Para continuar, em qualquer das máquinas digite CONT).

Vamos dar uma olhada no que está na tela (figura 2).

TRS-80 (MOD. I)

17129	244	mB	mB + MB * 256 = endereço do começo da próxima linha
17130	66	MB	
17131	10	mB	mB + MB * 256 = nº desta linha
17132	0	MB	
17133	147		REM (comprimido)
17134	48	0	
17135	48	0	
17136	48	0	
17137	48	0	
17138	42	*	
17139	0	← CÓDIGO DE FIM DE LINHA	

caracteres ⇒ / ⇐ códigos ASCII

PRÓXIMA LINHA

TK

16509	0	MB	MB * 256 + mB = nº desta linha
16510	10	mB	
16511	7	mB	mB + MB * 256 + 4 = nº de bytes ocupados por esta linha
16512	0	MB	
16513	234		REM (comprimido)
16514	0	0	
16515	0	0	
16516	0	0	
16517	0	0	
16518	23	*	
16519	118	← CÓDIGO DE FIM DE LINHA	

PRÓXIMA LINHA

Figura 2

Como você pôde observar , as linhas de programa começam a ser armazenadas, no TK, a partir do endereço 16509 e no TRS-80 (modelo I) a partir 17129. Os primeiros quatro bytes, em ambos os casos, referem-se à linha de programa. (Note as diferenças na figura). O código 118, no TK, indica o fim de linha. No TRS-80 esse código é Ø. O fim de programa é indicado pelo código repetido duas vezes (118, 118 no TK e Ø,Ø no TRS-80).

Dois desses bytes referem-se ao número de linha (veja figura). Portanto você pode alterá-lo, se quiser. Experimente:

POKE 1651Ø, 9 (para TK)
POKE 17131, 9 (para TRS-80)

E depois liste, o programa. A linha REM mudou de número. Mas cuidado porque se os números de linha ficarem fora de ordem, as coisas podem se complicar . . . Esta maneira de alterar números de linha (através de POKES) é utilizada nos renumeradores em Basic (Veja a Microhobby nº 1 ou a nº X). Você pode desenvolver seu próprio renumerador utilizando estas informações ou alterar o seu gosto o que nós já publicamos. Os outros dois bytes referem-se no TRS-80, ao endereço da próxima linha; no TK referem-se ao comprimento da linha. Se você der uns POKEs por aí, o seu programa provavelmente vai embaralhar . . . (sugestão: experimente, mas só depois de ter gravado o programa . . .) mais uma experiência interessante: localize na tela o código de fim de linha do seu computador. (Ø ou 118). Veja em que endereço está. Agora, no ***endereço seguinte,*** através de um POKE, coloque o ***mesmo*** código. No nosso caso, para a primeira linha, o POKE ficaria assim:

POKE 1652Ø, 118 (para o TK)
POKE 1714Ø, Ø (para o TRS-80)

Agora liste o programa. Só ficou a linha REM, certo? Isto porque o computador considera como indicação de fim de programa esses dois códigos repetidos!

Note que os comandos são comprimidos em um único byte e depois decodificados pela ROM na hora de apresentá-los na tela. Isto fica evidente para o TK mas não para o TRS-8Ø. (Veja de novo a figura 2), uma vez que a ROM só decodifica e apresenta na tela o comando descomprimido na apresentação da listagem. (Exemplo no TK, a instrução PRINT CHR$ (245) mostra na tela o comando PRINT. No TRS-80 não se conseguiria isto porque as instruções PRINT só se referem aos caracteres alfanuméricos e gráficos e nunca a ***comandos.*** Mas é possível mudar o REM da linha 1Ø. Experimente:

POKE 17133, 178 (para o TRS-80)
POKE 16513, 245 (para o TK)

Agora liste o programa novamente. A linha rem não é mais REM, é um PRINT! A ROM decodificou o byte pokeado naquele endereço como um comando, que evidentemente funcionará durante o programa. (Aqui a sugestão sem compromisso: que tal você escrever um programa que escreva outros programas? O caminho da construção das linhas é este . . .) ○

---

*Nota: Se você tem um TRS-80 modelo III (sem disco) ou compatível, modifique o endereço 17129 para 17386 e some 257 a todos os endereços.

LABIRINTO

Pierluigi Piazzi

Aristóteles considerava o cérebro humano como um órgão destinado simplesmente a esfriar o sangue. Só em meados do século XX as funções do sistema nervoso central começaram a ser parcialmente compreendidas.

Em 1861, o francês Paul Broca, ao analisar cérebros de cadáveres de pessoas que, em vida, tinham perturbações na fala, identificou, uma circunvolução no hemisfério esquerdo, responsável pelo processamento desta função.

Podemos considerar esta data como o início da decifração do HARDWARE da nossa mente. Esta decifração foi muito lenta e ainda hoje, um século depois, podemos nos considerar no começo.

Os estudos orientaram-se muito em direção ao *software* do nosso cérebro: a psicologia, a psicanálise e a psicolingüística.

Só muito recentemente, porém, é que a neuro-anatomia e a neurofisiologia começaram a se tornar as vedetes do progresso científico.

Um dos fatos mais importantes estabelecidos até agora, e cujo estudo rendeu um prêmio Nobel ao Dr. Sperry, é o da assimetria funcional do cérebro.

Anatomicamente os dois hemisférios cerebrais são praticamente iguais (a rigor um é quase a imagem especular do outro). Funcionalmente, porém, eles exercem papéis distintos:

O hemisfério esquerdo é responsável pela linguagem (nele se situa a ''zona de Broca''), pelo raciocínio aritmético e algébrico.

O hemisfério direito, ao contrário, é responsável pela percepção musical e espacial.

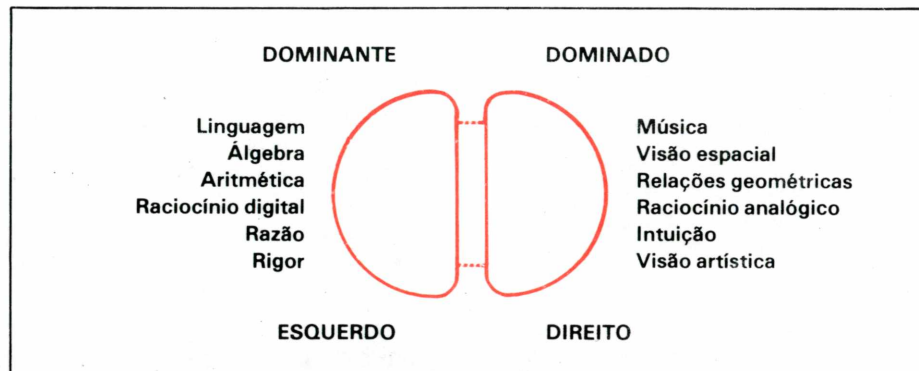
É como se tivéssemos, dentro do crânio, um cientista e um artista. Como o mesmo corpo não pode ser comandado por duas mentes, um dos dois hemisférios deve ''comandar''. Em noventa por cento das pessoas, o hemisfério dominante é o esquerdo.

Esta dominância reflete-se também socialmente: a ''razão'' ou seja, o raciocínio lógico-dedutivo é sempre levado mais a sério que a ''intuição''.

Mesmo que a nossa mente direita chegue a alguma conclusão verdadeira, a nossa mente esquerda sempre cobra como um ''demonstre''! Ora, a demonstração nada mais é que uma seqüência verbal, coisa que o nosso lado direito é incapaz de efetuar.

Em alguns países, entre os quais o Brasil, a dominância do lado esquerdo é culturalmente tão acentuada, que para quem acha esta afirmação um pouco radical, proponho que medite sobre alguns destes fatos:

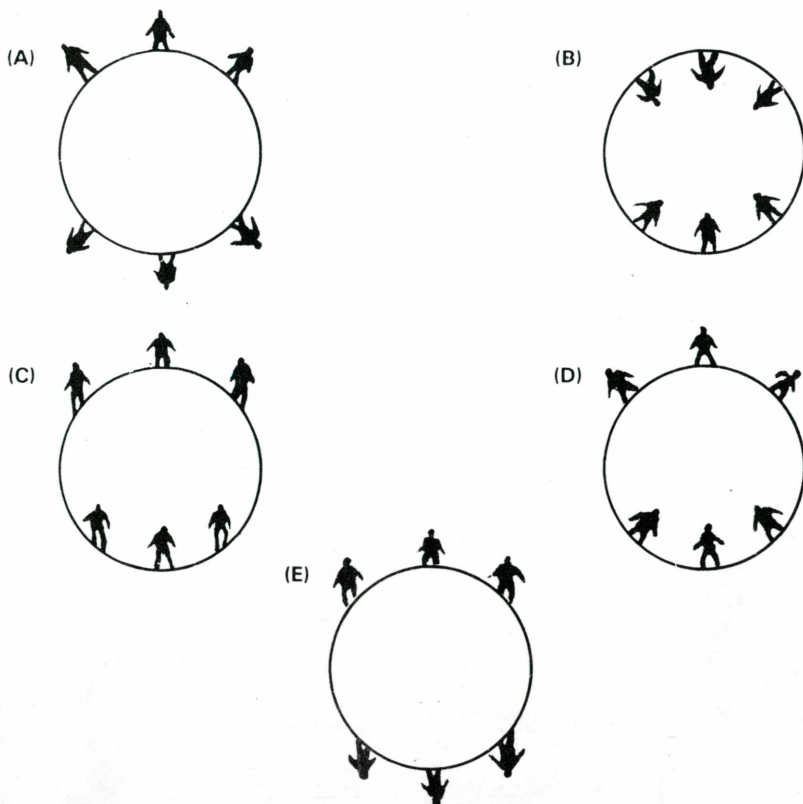
Pergunte por exemplo, a um transeunte nos Estados Unidos a indicação de algum logradouro: é extremamente comum se ouvir uma resposta do tipo: ''ande tantas quadras para leste e depois tantas para o norte''. Experimente perguntar à qualquer pessoa no Brasil, para que lado fica o Norte. Você terá sorte se ela souber distinguir a esquerda da direita. Ou então experimente mostrar a planta de um apartamento a várias pessoas. É surpreendente a quantidade das que não conseguem relacionar a planta com o apartamento real.

Na minha experiência como professor, duas coisas me chamaram par-

A especialização do cérebro humano.

ticularmente a atenção para este tipo de deficiência: uma delas foi a extrema dificuldade que sempre encontrei para ensinar determinados tópicos do eletromagnetismo, justamente os que exigem do aluno maior visão espacial. A outra é relativa a uma pergunta que me foi feita várias vezes: "se a água sempre corre para baixo, como é que tem rios que correm para o norte?"

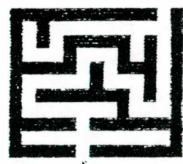
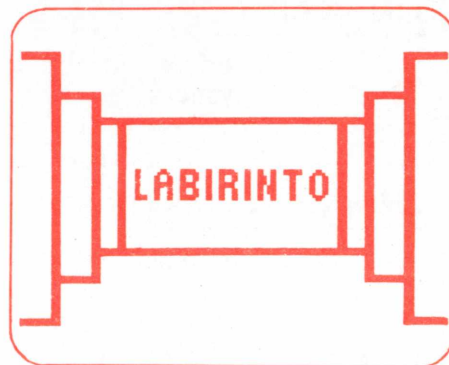
Na primeira vez que me fizeram esta pergunta, fiquei perplexo, depois percebi o motivo da confusão: como os mapas são pendurados na parede com o Norte para cima, na cabeça de muita gente um rio que corre para o norte está subindo! Este tipo de confusão é análogo à que faz com que muitos achem estranho que os japoneses, do outro lado do mundo, não caiam! O que a escola faz para evitar este tipo de ignorância espacial? Nada! Ensina Geometria de maneira axiomática e não intuitiva. Retira do vestibular a prova de Desenho (principalmente descritiva) ensina Geografia em mapas pendurados nas paredes, ou seja, não só se preocupa com isto como toma medidas que mostram como este tipo de problema é completamente ignorado.

Na primeira fase do exame vestibular deste ano da CESGRANRIO foi apresentada a seguinte questão:

(CESGRANRIO — 84) Qual das figuras abaixo melhor representa a situação dos habitantes da Terra em relação à mesma? As figuras ***não*** estão na escala.

Quando esta questão foi apresentada a um grupo de professores de Física de São Paulo, ela provocou os mais espantados comentários. Todos acreditaram que fosse impossível errar, entretanto, quando apresentamos a questão a um grupo de alunos ficamos surpresos com a porcentagem que assinalou ***b*** ou ***d***.

Pesquisando instrumentos e técnicas que permitam suprir esta deficiência, analisei o programa "Labirinto", comercializado pela Multisoft para TK 85, CP 200 ou TK 82-C e TK 83 com expansão para 16K.

Uma vez carregado o programa (4 minutos) ele gera a seguinte situação: Inicialmente, pergunta ao usuário o tamanho do labirinto (que pode ir de um mínimo de 3x3 células a um máximo de 15x9).

A seguir, se o usuário quiser, mostra, durante um certo tempo a ***planta*** do labirinto.

Em seguida, (esta é a parte mais bonita do programa), ele mostra o labirinto em três dimensões, ou seja, mostra uma perspectiva esquemática. Esta perspectiva corresponde ao ponto de vista de alguém adentrando no labirinto.

O usuário pode escolher entre virar para qualquer um dos outros três lados (para trás, à direita e à esquerda) ou dar um passo à frente.

```
PARA MOVER-SE UTILIZE JOYSTICK
     OU AS TECLAS ABAIXO:

  "7" - UM PASSO PARA FRENTE
  "5" - VIRA E OLHA A ESQUERDA
  "8" - VIRA E OLHA A DIREITA
  "6" - VIRA E OLHA PARA TRAS
  "0" - PEDE POR AJUDA
  "D" - DESISTO

VOCE QUER VER O PLANO DO
LABIRINTO ANTES DE ENTRAR NELE?

(S/N)
```

Se está perdido, pede ajuda e a tela mostra, durante algum tempo, a planta com a posição do usuário marcada por um asterisco.

O fundamental neste programa é a possibilidade de associar a planta bidimensional, vista de cima, com a localização em perspectiva simulando as três dimensões (se algum leitor quiser de aprofundar um pouco mais no estudo de hardware da mente humana, aconselharia o magnífico livro "Os Dragões do Éden" do astrônomo Carl Sagan — Ed. Francisco Alves).

Testando este programa com pessoas das mais variadas formações e idades, percebi duas coisas importantes: inicialmente ele permite detectar deficiências na visão espacial do usuário de maneira bem nítida. Posteriormente, com o uso continuado do "jogo" há uma melhora na habilidade do usuário, mostrando o caráter didático do programa.

Resumindo, esta fita, no mercado há bastante tempo, tem sido comercializada (e comprada) como jogo, quando na realidade se constitui num instrumento didático da mais alta qualidade. ○

ATAQUE ESPACIAL

Gustavo Egídio de Almeida

Memória — 1445 bytes
Soma Sintática — 34740

Você dispõe de 200 Tiros de Laser e necessita abater no mínimo 15 naves invasoras. Para alcançar o final do jogo.

```
*************************
VOCE VENCEU
ENERGIA RESTANTE: 195
```

Caso você deixe passar acima de 5 naves sem o alcance de seu laser, perderá o jogo e será destruído.

Para mover sua nave use as seguintes teclas:

6 — Move sua nave para baixo
7 — Move sua nave para cima
8 — Dispara o Laser

```
  10 LET A=PI-PI
  20 LET J=VAL "200"
  30 LET K=VAL "10"
  40 LET G=VAL "0"
  50 LET X=INT (RND*VAL "18")+VA
L "2"
  60 LET A=A+VAL "1"
  70 IF A=VAL "21" THEN GOTO VAL
 "260"
  80 LET Y=VAL "30"
  90 PRINT AT K,VAL "0";CHR$ VAL
 "130";CHR$ VAL "146";AT X,Y;"X"
 100 IF J<A THEN GOTO VAL "150"
 110 IF INKEY$="7" THEN LET K=K-
VAL "1"
 120 IF INKEY$="6" THEN LET K=K+
VAL "1"
 130 IF INKEY$="8" THEN PRINT AT
 K,VAL "2";"**************************
"
 140 IF INKEY$>"5" THEN LET J=J-
VAL "1"
 150 LET Y=Y-VAL "1.5"
 160 IF Y=VAL "3" THEN LET G=G+V
AL "1"
 170 IF G=VAL "5" THEN GOTO VAL
"240"
 180 IF Y=VAL "3" THEN GOTO VAL
"50"
 190 IF INKEY$="8" AND K=X AND Y
<21 THEN GOTO VAL "220"
 200 CLS
 210 GOTO 90
 220 PRINT AT X,Y+VAL "1";CHR$ V
AL "189"
 230 GOTO VAL "50"
 240 PRINT "VOCE FOI DESTRUIDO"
 250 STOP
 260 PRINT "VOCE VENCEU"
 270 PRINT "ENERGIA RESTANTE: ";
J
```

TAENIA KOMPUTADORUM

Taenia Komputadorum
Memória Ocupada = 1866 bytes
Soma Sintática = 1053
Nível =
Computadores = TK-83 e compatíveis

Renato da Silva Oliveira

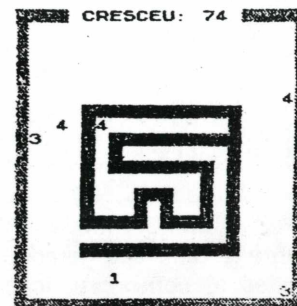
Sei que é difícil acreditar, caros leitores, mas seu TK redigiu sozinho o texto acima! Eu apenas tirei uma cópia da tela onde o micro o produziu, para poder apresentá-lo a vocês! Isto ocorreu por volta das 4:30 da madrugada do dia em que eu rodei nele, pela primeira vez, o programa listado na figura 1. Eu havia passado a noite inteira brincando com ele e, quando os primeiros clarões avermelhados do sol cingiam o horizonte leste prenunciando o raiar de um novo dia, entre os últimos dois goles de um jarro de suco de morangos silvestres que ganhei de meu amigo Nabor Rosenthal, fui surpreendido com a mensagem na tela.

Após lê-la, senti-me muito mal; digitei COPY (e N/L) e adormeci!

Depois disso venho tentando, sem êxito, obter outras mensagens. Na esperança de que algo semelhante aconteça com vocês, resolvi comunicar-lhes o que me ocorreu. Se algo estranho acontecer com seus TK's, por favor, não deixem de nos informar.

O programa Taenia Komputadorum cabe com apenas dois kbytes de RAM e você dirige o verme através das teclas 5, 6, 7 e 8 (ou joystick).

```
   0▒REM "▒AENIA ▒OMPUTADORUM"
  10 FAST
  20 FOR A=PI-PI TO VAL "21"
  30 PRINT AT A,PI-PI;"▒";TAB VA
L "21";"▒";AT PI-PI,A;"▒";AT VAL
 "21",A;"▒"
  40 NEXT A
  50 LET L=VAL "11"
  60 LET C=L
  70 LET Y=L
  80 LET X=C
  90 LET A$=""
 100 LET C$=""
 110 LET Z=PI/PI
 120 LET B=PI-PI
 130 LET W=B
 140 SLOW
 150 PRINT AT Y,X;"▒"
 160 IF W>VAL "15" THEN GOTO 250
 170 LET S=INT (PI/PI+VAL "5"*RN
D)
 180 LET T=PI/PI+VAL "20"*RND
 190 LET U=PI/PI+VAL "20"*RND
 200 PRINT AT T,U;
 210 GOSUB VAL "490"
 220 IF E<>PI-PI THEN GOTO VAL "
180"
 230 PRINT S
 240 LET W=W+S
 250 LET D$=INKEY$
 260 IF D$<>"" THEN LET C$=D$
 270 IF C$<>"5" AND C$<>"6" AND
C$<>"7" AND C$<>"8" THEN GOTO VA
L "250"
 280 LET A$=A$+C$
 290 LET Y=Y+(PI/PI AND C$="6")-
(PI/PI AND C$="7")
 300 LET X=X+(PI/PI AND C$="8")-
(PI/PI AND C$="5")
 310 PRINT AT Y,X;
 320 GOSUB VAL "490"
 330 IF E=PI-PI THEN GOTO VAL "4
00"
 340 IF E<>VAL "8" AND E<>VAL "1
28" THEN GOTO VAL "370"
 350 PRINT AT PI-PI,VAL "4";" CR
ESCEU: ";Z-PI/PI;" "
 360 STOP
 370 LET B=B+E-VAL "28"
 380 LET Z=Z+E-VAL "28"
 390 LET W=W-E+VAL "28"
 400 PRINT "█"
 410 IF B>PI-PI THEN GOTO VAL "4
70"
 420 PRINT AT L,C;" "
 430 LET L=L+(PI/PI AND A$(1)="6
")-(PI/PI AND A$(1)="7")
 440 LET C=C+(PI/PI AND A$(1)="8
")-(PI/PI AND A$(1)="5")
 450 IF LEN A$>PI-PI THEN LET A$
=A$(VAL "2" TO )
 460 GOTO VAL "150"
 470 LET B=B-PI/PI
 480 GOTO VAL "150"
 490 LET E=PEEK (PEEK 16398+256*
PEEK 16399)
 500 RETURN
```

Seu objetivo é alimentá-lo com a maior quantidade de comida (números inseridos aleatoriamente na tela) possível. Cada vez que a Taenia é alimentada, ela cresce tanto quanto o **valor nutritivo** de sua comida. Cuidado para não fazê-la perfurar as **paredes intestinais** do micro (quadrado na tela), ou ainda, devorar a si mesma. Caso isto ocorra, a Taenia morre e surge na tela o comprimento que ela cresceu sendo alimentada por você (figura 2).

A Taenia morre comendo a si mesma . . .

Neste mês, apresentamos um programa muito especial para você que gosta de romances e que é um eterno sonhador: Romeu e Julieta, a famosa peça de teatro de Shakespeare.

"Romeu e Julieta" são dois jovens apaixonados, cujas famílias são inimigas e não permitem, de maneira alguma, o romance entre os dois. Mas eles se amam e não se intimidam pelas ameaças de seus pais.

Julieta pensa numa solução para libertar-se de vez das amarras paternas: fugir com Romeu. Para isto, ela procura o sempre bom amigo padre da família para auxiliá-la e este lhe entrega um frasco com uma poção que, ao tomá-la, Julieta ficará adormecida e todos pensarão que ela esta morta. Julieta será colocada no mausoléu da família e Romeu irá buscá-la.

Porém, nada dá certo no plano da bela Julieta! Não avisam Romeu a tempo e o jovem, desesperado, corre ao mausoléu e encontra sua amada aparentemente morta!

O apaixonado Romeu desespera-se e se mata com sua espada. Nisso, Julieta renasce de seu sonho e vê seu amado morto ao seu lado. A bela jovem percebe neste momento, que o único caminho para ficarem juntos, é a morte . . .

Assim termina o trágico romance de Romeu e Julieta! . . . e inicia-se o nosso jogo-aventura . . .

Romeu e Julieta é um programa para ser usado no seu TK, com expansão de 16k ou mais de memória. Um típico jogo-aventura que funciona basicamente como um programa educativo, ou seja, um programa cuja finalidade é desenvolver o aprendizado de jovens na faixa infanto-juvenil (figura 1).

```
ROMEU E JULIETA

ESTE E' UM MATEMATICO
JOGO-AVENTURA.

SERAO GERADOS UM CERTO NUMERO DE
PERGUNTAS DURANTE AS QUAIS
NOTAR-SE-A' O DESENROLAR DO
FAMOSO ROMANCE SHAKESPERIANO.
```

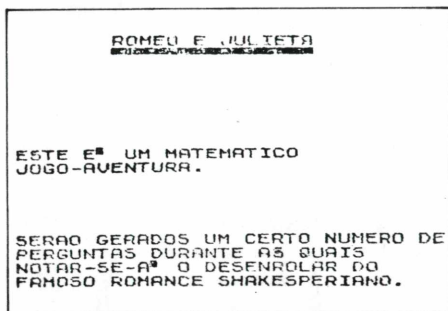



Figura 1

Durante o jogo, são gerados inúmeros testes e, como gratificação a cada resposta certa, o computador se encarrega de produzir uma cena com efeitos de animação: uma cena da história é simulada na tela e sua seqüência é conseguida através das respostas certas inseridas no computador.

Quando o programa é posto para rodar, após uma rápida explicação do funcionamento do jogo, surge, no canto inferior da tela, o cursor L, indicando que o computador está pedindo dados.

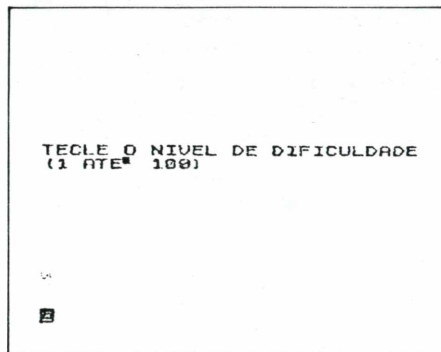
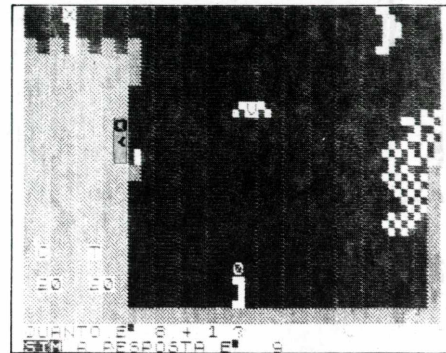
Figura 2

Estes dados referem-se ao nível de dificuldade do jogo que pode variar de 1 até 100 (figura 2). Em qualquer um destes níveis, são geradas questões envolvendo as quatro operações aritméticas: soma, subtração, divisão e

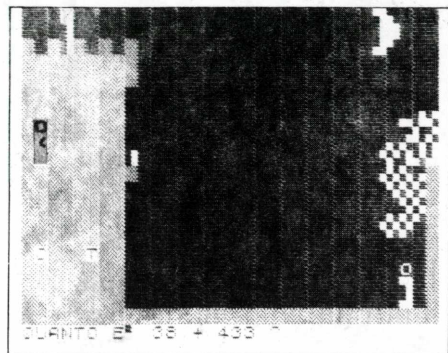
Figura 3

ROMEU E JULIETA
Memória Ocupada = 7504 bytes

Nível =

Computadores = TK-83 e compatíveis

multiplicação.

No nível 1, são propostas operações com um ou dois dígitos (figura 3). No nível 100, ou seja, no nível de dificuldade mais elevado, são geradas operações com três dígitos (figura 4).

Figura 4

São geradas um total de 40 questões, no máximo.

À cada questão correta, notar-se-á mudanças no cenário apresentado na tela.

A seguir, citaremos algumas das cenas desenvolvidas na tela:

— A jovem Julieta move-se em direção à da torre. Romeu, movimenta-se em direção ao castelo e, no meio de seu percurso, dois fatos ocorrem: Julieta, avistando seu amado, joga uma flor em sua direção, como prova de seu amor. Uma coruja, que estava no topo da torre, percorre o céu de um canto ao outro da tela e pousa numa árvore em frente ao castelo.

— Romeu, ao chegar ao pé da torre, inicia uma difícil mas compensadora escala até a sacada da torre.

Por fim, Romeu abraça Julieta, despedindo-se logo em seguida.

Julieta recolhe-se aos seus aposentos e Romeu inicia uma caminhada de volta ao ponto de origem.

Basicamente, são essas as cenas demonstradas na tela, no decorrer do jogo.

Por fim, caso o jogo for bem sucedido, o computador apresenta no vídeo a seguinte mensagem:

"Romance concluído, tecle Z para recomeçar". Caso contrário, Romeu morrerá e surgirá na tela a mensagem:

"Que azar, hein? Tecle Z para recomeçar". →

Programa adaptado e reestruturado da revista Popular Computing Weekly — Março/83.

PROGRAMA DO MÊS

Gustavo Egídio de Almeida

ROMEU E JULIETA

```
   5 REM ROMEU E JULIETA
  10 PRINT AT 4,7;"ROMEU E JULIE
TA";AT 5,7;"████████████████"
  20 PRINT AT 10,0;"ESTE E' UM M
ATEMATICO            JOGO-AVENTUR
A.";AT 15,0;"SERAO GERADOS UM CE
RTO NUMERO DEPERGUNTAS DURANTE A
S QUAIS          NOTAR-SE-A' O DESEN
ROLAR DA         FAMOSA TRAGEDIA SHA
KESPERIANA."
  30 FOR F=1 TO 400
  31 NEXT F
  40 CLS
  50 PRINT AT 10,0;"TECLE O NIVE
L DE DIFICULDADE      (1 ATE' 100)
"
  60 INPUT P
  70 IF P<1 OR P>100 THEN GOTO 4
0
  80 CLS
 110 LET A$="████████████████████
████████████"
 120 LET B$="████████"
 130 LET C$="████████"
 140 LET D$="████████"
 150 LET E$="█████████"
 160 LET F$="████████████████████
████████████"
 170 LET G$="
                                  "
 180 FOR E=0 TO 18
 190 PRINT AT E,0;A$
 200 NEXT E
 210 PRINT AT 2,0;"██████████";AT
3,0;E$;AT 4,0;E$;AT 10,0;E$
 220 FOR E=5 TO 9
 230 PRINT AT E,0;D$
 240 NEXT E
 250 FOR E=11 TO 17 STEP 2
 260 PRINT AT E,0;C$;AT E+1,0;B$
 270 NEXT E
 280 PRINT AT 19,0;F$
 310 PRINT AT 6,30;"██";AT 7,29;
"███";AT 8,28;"████";AT 9,27;"██
██";AT 10,27;"█████"
 320 PRINT AT 11,28;"████";AT 12
,29;"███";AT 13,27;"█████";AT 14
,28;"████"
 330 FOR E=15 TO 18
 340 PRINT AT E,31;"█"
 350 NEXT E
 360 PRINT AT 15,1;"C";TAB 5;"T"
 410 LET K=7
 420 LET J=1
 430 LET S=16
 440 LET R=29
 450 PRINT AT K,J;"O";AT K+1,J;"
<";AT K+2,J;" "
 460 PRINT AT S,R;"█";AT S+1,R;"
█ ";AT S+2,R;"█ "
 470 PRINT AT 0,27;"██";AT 1,27;
"█ ";AT 2,27;"██"
 480 PRINT AT 9,8;"█";AT 0,3;"X"
;AT 1,3;"█";AT 2,3;"█"
 900 LET N=0
 910 LET TOT=0
1010 LET A=INT (RND*10*P)
1020 LET B=INT (RND*5*P)
1030 LET C=INT (RND*10*P)
1040 LET F=INT (RND*5*P)+1
1050 LET G=INT (RND*2*P)
1060 LET H=A+B
1070 LET I=C-F
1080 LET M=B*F
1090 LET O=A*G
1100 PRINT AT 20,0;G$;AT 21,0;G$
1110 PRINT AT 20,0;"QUANTO E' ";
A;" + ";B;" ?"
1120 INPUT X
1130 IF X<>H THEN PRINT AT 21,0;
"███ A RESPOSTA E'   ";H
1140 IF X=H THEN PRINT AT 21,0;"
███ A RESPOSTA E'   ";H
1150 IF X=H THEN GOSUB 4000
1160 GOSUB 4500
1200 PRINT AT 20,0;G$;AT 21,0;G$
1210 PRINT AT 20,0;"QUANTO E' ";
C;" - ";F;" ?"
1220█INPUT X
1230 IF X<>I THEN PRINT AT 21,0;
"███ A RESPOSTA E'   ";I
1240 IF X=I THEN PRINT AT 21,0;"
███ A RESPOSTA E'   ";I
1250 IF X=I THEN GOSUB 4000
1260 GOSUB 4500
1300 PRINT AT 20,0;G$;AT 21,0;G$
1310 PRINT AT 20,0;"QUANTO E' ";
M;" / ";F;" ?"
1320 INPUT X
1330 IF X<>B THEN PRINT AT 21,0;
"███ A RESPOSTA E'   ";B
1340 IF X=B THEN PRINT AT 21,0;"
███ A RESPOSTA E'   ";B
1350 IF X=B THEN GOSUB 4000
1360 GOSUB 4500
1400 PRINT AT 20,0;G$;AT 21,0;G$
1410 PRINT AT 20,0;"QUANTO E' ";
A;" X ";G;" ?"
1420 INPUT X
1430 IF X<>O THEN PRINT AT 21,0;
"███ A RESPOSTA E'   ";O
1440 IF X=O THEN PRINT AT 21,0;"
███ A RESPOSTA E'   ";O
1450 IF X=O THEN GOSUB 4000
1460 GOSUB 4500
1500 GOTO 1000
4010 IF N<=5 THEN GOSUB 5000
4020 IF N=6 THEN GOSUB 5500
4030 IF N>6 AND N<11 THEN GOSUB
6000
4040 IF N=11 THEN GOSUB 6500
4050 IF N>11 AND N<29 THEN GOSUB
 6000
4060 IF N=20 THEN GOSUB 7000
4070 IF N>=29 AND N<34 THEN GOSU
B 7500
4080 IF N=34 THEN GOTO 8000
4090 LET N=N+1
4100 PRINT AT 17,1;N
4200 RETURN
4510 LET TOT=TOT+1
4520 IF TOT=40 THEN GOTO 8500
4530 PRINT AT 17,5;TOT
4540 RETURN
5010 PRINT AT K,J;"█O";AT K+1,J;
"█<";AT K+2,J;"█ "
5020 IF J=6 THEN GOSUB 5500
5030 LET J=J+1
5040 RETURN
5510 LET Z$="████████████"
5520 PRINT AT 6,10;"ROMEU,ROMEU"
;AT 7,10;"ONDE";AT 8,10;"TU ESTA
S";AT 9,10;"ROMEU  ?  "
5530 GOSUB 9990
5540 PRINT AT 6,10;Z$;AT 7,10;Z$
;AT 8,10;Z$;AT 9,10;Z$
5550 RETURN
6010 LET R=R-1
6020 PRINT AT S,R;"██";AT S+1,R;
"██";AT S+2,R;"██"
6030 RETURN
6510 LET X=400*COS (PI/3)
6520 LET Y=400*SIN (PI/2)
6530 FOR Z=8 TO Y/16
6540 LET W=.01*(Y*Z-16*Z*Z)
6550 UNPLOT .01*X*Z,W+10
6570 PLOT .01*X*Z,W+10
6580 NEXT Z
6590 RETURN
7010 FOR D=9 TO 26
7020 PRINT AT 6,D;"██V█";AT 6,D;
"██V█"
7030 NEXT D
7040 PRINT AT 6,27;"██ ";AT 5,29
;"V"
7050 RETURN
7510 LET S=S-1
7520 PRINT AT S,R;"█";AT S+1,R;"
█ ";AT S+2,R;"█ ";AT S+3,R;"█"
7530 RETURN
8010 PRINT AT 13,8;"█";AT 12,8;"
█"
8020 PRINT AT 11,8;"█ ";AT 10,9;
"█"
8030 GOSUB 9900
8040 PRINT AT 11,8;"██";AT 8,9;"
█";AT 9,9;"█ ";AT 10,9;"█ "
8050 GOSUB 9900
8060 PRINT AT 10,9;"█";AT 9,9;"█
 ";AT 8,9;"█ ";AT 7,9;"█"
8070 GOSUB 9900
8080 PRINT AT 7,8;"██";AT 8,8;"█
█";AT 9,8;"██"
8090 GOSUB 9900
8100█PRINT AT 6,10;"BOA NOITE";A
T 7,10;"BOA NOITE";AT 8,10;"TENH
O QUE";AT 9,10;"PARTIR AGORA";AT
 10,10;"NAO FIQUE MAGOADA"
8110 GOSUB 9990
8120 LET P$="██████████████████"
8130 PRINT AT 6,10;P$;AT 7,10;P$
;AT 8,10;P$;AT 9,10;P$;AT 10,10;
P$
8150 LET J=J-1
8160 PRINT AT K,J;"O█";AT K+1,J;
">█";AT K+2,J;" █"
8170 IF J=0 THEN GOTO 8200
8190 GOTO 8150
8200 PRINT AT 7,8;"██";AT 8,8;"█
";AT 9,8;"██"
8210 GOSUB 9900
8220 PRINT AT 7,9;"█";AT 8,9;"█"
;AT 9,9;"█ ";AT 10,9;"█ "
8230 GOSUB 9900
8240 PRINT AT 8,9;"█";AT 9,9;"█"
8250 PRINT AT 11,8;"█ ";AT 10,9;
"█"
8260 GOSUB 9900
8270 PRINT AT 10,9;"█";AT 11,8;"
██"
8280 FOR S=11 TO 15
8290 PRINT AT S,8;"█";AT S+1,8;"
█";AT S+2,8;"█ ";AT S+3,8;"█ "
8300 NEXT S
8310 FOR R=8 TO 29
8320 PRINT AT 16,R;"██";AT 16+1,
R;"██";AT 16+2,R;"██"
8330 NEXT R
8340 LET Q$="                    "
8350 FOR E=6 TO 12
8360 PRINT AT E,10;Q$
8370 NEXT E
8380 PRINT AT 6,10;" ROMANCE CON
CLUIDO ";AT 8,10;"TECLE █ PARA";
AT 10,10;"RECOMECAR"
8390 INPUT Z$
8400 IF Z$="Z" THEN GOTO 100
8410 GOTO 8390
8510 LET D=26
8520 PRINT AT S+1,D;"█";AT S+1,D
+1;"█"
8530 LET D=D-1
8540 IF D=R THEN GOTO 8600
8545 IF D<10 THEN GOTO 8730
8550 GOTO 8520
8600 GOSUB 9900
8610 PRINT AT S-1,R;"████";AT S,R
;"████";AT S+1,R;"█ █";AT S+2,R;"
███"
8620 IF S=16 THEN GOTO 8700
8630 LET S=S+1
8640 GOTO 3610
8700 PRINT AT 16,R;"████";AT 17,R
;"████";AT 18,R;"████"
8710 GOSUB 9900
8720 PRINT AT 7,7;"█";AT 8,7;"█"
;AT 9,7;"█"
8730 LET J=J-1
8740 PRINT AT K,J;"O█";AT K+1,J;
">█";AT K+2,J;" █"
8750 IF J=0 THEN GOTO 8800
8760 GOTO 8730
8800 LET Q$="                    "
8810 FOR E=6 TO 12
8820 PRINT AT E,10;Q$
8830 NEXT E
8840 PRINT AT 6,10;"QUE AZAR HEI
N?";AT 8,10;"TECLE █ PARA";AT 10
,10;"RECOMECAR"
8850 INPUT Z$
8860 IF Z$="Z" THEN GOTO 100
9000 GOTO 8850
9800 STOP
9900 FOR T=1 TO 10
9910 NEXT T
9920 RETURN
9990 FOR T=1 TO 40
9991 NEXT T
9992 RETURN
9998 SAVE "R/█"
9999 RUN
```

COMO FAZER SUA ASSINATURA

Para obter seu exemplar mensal (12 números) da revista **Microhobby** contendo muitos programas para o TK como também para o Apple e o TRS-80, inúmeras dicas e as últimas novidades na área de informática, você deve fazer uma assinatura.

É importante ressaltar que o recebimento da revista é considerado a partir da data de recebimento do pedido de assinatura, porém, há um período de 30 dias de ***carência*** até a revista chegar às suas mãos. Os números anteriores da revista podem ser adquiridos se as tivermos no momento do pedido em nossos estoques, através do telefone 255-0722, diretamente com o Departamento Comercial.

Para ter acesso a todas estas vantagens basta preencher, corretamente, o cupom anexo, colocá-lo no correio junto a um cheque nominal ou vale postal em nome de Micromega Publicações e Material Didático, no valor de Cr$ 16.900,00.

O envelope deverá ser selado e endereçado com os seguinte dizeres:

> **MICROMEGA P.M.D. LTDA.**
> **Departamento de Assinaturas**
> **Caixa Postal** 54096
> **CEP 01296 – São Paulo, SP.**

e dentro dele não deverá ter nada além do cheque e o cupom.

No verso do cheque, escreva:

"Destina-se ao pagamento de uma assinatura (12 números) da revista Microhobby".

Quando este cheque for devolvido ao seu Banco com nosso endôsso, servirá (para você), de comprovante provisório até que nosso recibo seja confeccionado e enviado pelo correio.

por dentro do apple

Prof. Wilson José Tucci — Coordenador de Projetos Especiais da Escola Experimental Pueri Domus.

MAIS SOBRE A MODIFICAÇÃO DE COMANDOS

Tivemos no número 7 uma introdução sobre como fazer alterações no *firmware* do Apple, utilizando como exemplo um programa que copiava o conteúdo da ROM para o cartão de expansão de memória, permitindo assim a modificação das palavras reservadas do Applesoft. Veremos aqui, como prometido, uma maneira de fazer o mesmo com os comandos de DOS, que podem ser alterados sem a necessidade de se ter um *language card* instalado.

O mesmo programa pode ser aproveitado, com algumas modificações, já que o DOS armazena suas palavras-chave no mesmo formato que o Applesoft. Ou seja, a palavra que corresponde a cada comando é guardada como uma cadeia contínua de caracteres, com o final de uma palavra indicado pelo bit 7 ligado em sua última letra. A diferença está, é claro, no endereço do início da lista e no número de palavras; como estes dados foram definidos como variáveis na linha 1ØØ do programa TRADUTOR DE PALAVRAS, esta é a primeira a ser alterada. A variável P (posição ou endereço inicial), deve ser inicializada para 4314Ø ($A884),* e a variável NP (número de palavras na lista) para 28.

A próxima linha a ser modificada é a linha 11Ø, apenas para corrigir a apresentação do programa. Como não vamos necessitar da rotina MOVE do monitor para passar o conteúdo da ROM para o *language card,* nem dos comandos que controlam esse cartão, a linha 24Ø (e portanto também a 27Ø) pode ser removida. A última alteração a ser feita está na linha 26Ø, que contém um POKE 4928Ø,Ø, cuja função era de "ativar" o cartão de 16k; este comando deve ser retirado do programa. Esta é realmente, a única alteração que pode causar problemas se não for feita porque pode fazer com que o computador fique "pendurado", de forma que, até a tecla RESET perca seu efeito.

O programa

Abaixo aparece novamente a listagem do programa, indicando as linhas que foram modificadas. Para quem perdeu o artigo da revista número 7 vamos ver, resumidamente, o funcionamento geral do programa.

As 28 palavras-chave do DOS estão armazenadas em uma lista a partir da posição 4314Ø ($A884), no formato descrito acima. As primeiras duas palavras da lista são INIT e LOAD, portanto a tabela mostra o conteúdo dos primeiros 8 bytes de memória:

Posição	Conteúdo	ASCII
4314Ø ($A884)	73 ($49)	I
43141	78 ($4E)	N
43142	73 ($49)	I
43143	212 ($D4)	T
43144	76 ($4C)	L
43145	79 ($4F)	O
43146	65 ($41)	A
43147	196 ($44)	D

O programa coloca no vetor A$, nos elementos de índices 1 a 28, a lista de palavras-chave retirada da memória pelo *loop* 12Ø-14Ø. A variável NB mantém o número de bytes livres (em relação ao tamanho original da lista), para que a lista não tenha o seu tamanho aumentado. A seguir o programa entra em outro *loop,* composto pelas linhas 15Ø a 22Ø, aonde uma palavra é pedida; a palavra com seu bit 7 ativado é procurada na lista e substituída por outra, digitada pelo usuário. Quando este responde com uma linha nula, digitando apenas RETURN, o programa faz a alteração final, com a aprovação do usuário, gravando os novos comandos caractere por caractere (linha 25Ø).

Fazendo modificações definitivas

Há basicamente quatro formas de se poder usar o DOS alterado sempre que o disco for "booteado". A primeira consiste em modificar o que for necessário na memória e depois inicializar um novo disco, no qual será gravada uma cópia deste DOS. O único problema é que este será um disco "escravo", ou seja, cujo DOS não é relocável; o disco não poderá ser usado para inicializar um sistema com menos de 48K (assumindo que o leitor tenha 48K ou mais em seu computador) ou não utilizará a memória eficientemente, quando "booteado" em Apples com maior capacidade de memória que aquele aonde foi inicializado. Se o programa MASTER CREATE for rodado sobre este disco, a versão modificada do DOS será perdida e substituída pela original.

Um segundo método é o de fazer as modificações, todas as vezes que o disco é "booteado", o que pode ser feito pelo programa HELLO ou aquele que é executado na fase de *boot*). Se o disco sempre for usado num sistema de 48K, os endereços podem ser fixos. No entanto, se o disco deve ser transportável, o HELLO deve verificar de alguma forma o tamanho da memória do computador; uma das várias formas de fazer isto é testar o vetor de re-entrada no DOS, localizado nas posições 976-978 ($3DØ-$3D2). Um PEEK (978) devolverá 157 para um sistema de 48K, ou 93 para um de 32K. O valor dessa função pode ser usado como constante no cálculo dos endereços internos do DOS, o que torna o programa seguro de se rodar em computadores com 24K, 36K, ou qualquer tamanho de memória que não é possível ter nas versões mais recentes do Apple.

Outra forma de tornar definitivas as modificações é fazê-las diretamente nos três primeiros *tracks,* reservados ao DOS, através de programas que permitem a manipulação do disco ao nível do setor. É aconselhável fazer isso com muito cuidado, porque qualquer descuido pode destruir a cópia do DOS nesse disco, tornando-se necessária a sua reconstituição, geralmente através do MASTER CREATE. Este método criará um disco "mestre" com o novo DOS, mesmo que esse disco já fosse "mestre" antes da modificação.

*** Observação: Para as referências a todos os endereços de DOS acima de 384ØØ ($96ØØ) foi assumido que o leitor possui um sistema com pelo menos 48K; para 32K, subtraia 16384 ($4ØØØ) do endereço dado. Todos os números hexadecimais são indicados por um $.**

O último método usa o programa MASTER CREATE para criar um disco "mestre" com qualquer modificação. Esse programa, ao ser executado, carrega uma imagem do DOS original na memória do Apple, procedendo depois para copiar esta imagem nos discos a serem atualizados. O programa pode ser interrompido logo após a imagem ser carregada, permitindo sua modificação, para depois prosseguir com o MASTER CREATE e copiar o DOS novo para todos os discos que se desejar. Para fazer isso, siga os seguintes passos:

— carregue o programa com um BLOAD MASTER CREATE;

— entre no monitor (CALL-151) e altere a posição $80D para $4C, com o comando 80D:4C;

— inicie a execução do programa através de um 800G e aguarde até que a imagem do DOS seja carregada e o programa pare;

— uma vez de volta ao monitor, use seus comandos para fazer as alterações desejadas, lembrando de subtrair $7F00 dos endereços de DOS (ou $3F00 para 32K), devido ao fato de que essa imagem é carregada a partir da posição $1200;

— feitas todas as alterações, entre novamente no MASTER CREATE através do comando 82DG; o procedimento a partir deste ponto é igual ao seguido para atualizar discos com o DOS normal, e a imagem modificada pode ser transferida a quantos discos for necessário.

Outras modificações possíveis

Entre as inúmeras modificações que podem ser efetuadas no DOS, podemos mencionar algumas:

Mensagens de erro

Nos últimos números já tínhamos falado sobre a possibilidade de alterarmos o texto das mensagens de erro do DOS. Isto, como também foi dito, pode ser feito de maneira semelhante ao método usado para alterar os comandos. De fato, podemos usar o mesmo método se o tamanho das *strings* que compõem o texto de cada mensagem não for alterado. Isso é devido ao fato de que cada mensagem tem seu local de início fixo na memória — um texto pode até ser diminuído em tamanho, desde que seu início não seja alterado. Mas, pensando bem, os endereços que indicam o início de cada mensagem estão numa tabela, e essa tabela está, como o resto do DOS, em RAM — e portanto pode ser modificada!

Na verdade, essa tabela fornece o *offset,* ou endereço relativo, a partir do início da lista de caracteres das mensagens. O texto das mensagens está no mesmo formato que o dos comandos, com o final de uma mensagem indicada por um byte com seu bit 7 alto. Essa lista está localizada não muito longe dos comandos, nas posições 43377 a 43583 e 43538 ($AA3F e $AA4E). Como o primeiro byte da tabela é 0, o início da primeira mensagem está em 43377: sendo o segundo 4 significa que a segunda mensagem tem seu início na posição 43377 + 4, ou seja, em 43381.

Outros textos

Que textos? O DOS está cheio dos mais variados tipos de texto, com os quais o usuário comum muitas vezes nunca se depara. Isso, é claro, não ocorre com todos os textos; alguns dos que podem ser interessantes mudar são:

— os símbolos que indicam o estado (*locked* ou *unlocked*) de um arquivo no catálogo; os símbolos originais ("*" e " ") estão armazenados respectivamente nas posições 44515 ($ADE3) e 44508 ($ADDC);

— outros símbolos apresentados no catálogo são os que indicam o tipo de cada arquivo, que são oito (mas atualmente os quatro últimos não têm significado definido, o que não impede de serem usados por alguns); o *string* "TIABSRAB" está armazenado da posição 45991 a posição 45998 ($B3A7 — $B3AE);

— ainda nos textos produzidos durante um CATALOG, temos a mensagem de volume do disco ("DISK VOLUME"), que ocupa os 12 bytes a partir de 45999 ($B3AF) armazenado em ordem inversa;

— outra área de textos que pode ser modificada é a que armazena as palavras-chave que antecedem os parâmetros dos comandos de DOS; a *string* "VDSLRBACIO" está gravada com os MSBs no estado alto a partir de 43329 ($A941);

— o caractere de comando do DOS (ctr1-D) está armazenado na posição 43698 ($AAB2); este pode ser mudado para qualquer outro, inclusive caracteres visíveis.

Alterações de software

Ao contrário do que foi feito até agora, modificando apenas textos impressos durante a execução de algumas rotinas do DOS, as alterações descritas abaixo mudam a própria execução dessas rotinas:

— o RUN que é executado na última fase de *boot* pode ser mudado para BRUN ou até EXEC, a fim de podermos usar como HELLO um programa em linguagem de máquina ou um arquivo-texto; para tanto, basta substituir o 6 na posição 40514 ($9E42) por um 52 ($34) para BRUN ou 20 ($14) para EXEC;

— a pausa durante a apresentação de um catálogo muito longo pode ser retirada, se desejado, colocando-se um RTS (96 ou $60) na posição 44596 ($AE34);

— os nomes de arquivos deletados podem ser apresentados durante um catálogo (desde que não tenha entrado outro nome sobre esse) colocando-se NOPs (234 ou $EA) nas posições 44505 e 44506 ($ADD9 e $ADDA);

— inversamente, se você colocar NOPs nas posições 44599 e 44600 ($AE37 e $AE38), a rotina de apresentação do catálogo colocará uma pausa depois de cada nome de arquivo;

— mais uma com a instrução de operação nula: se forem colocados três NOPs a partir da posição 49107 ($BFD3) o DOS não forçará o recarregamento do *language card* se já havia uma linguagem presente no cartão (apenas o DOS 3.3 faz isso — as versões anteriores não precisam ser modificadas);

— se você não quer mensagens de erro (de DOS, é claro), coloque NOPs nas posições 42768 a 42770 ($A710 a $A712).

Observações

Há infinitas alterações possíveis de serem feitas no DOS — estas foram apenas algumas. O objetivo deste artigo foi apenas o de introduzir os leitores interessados a alguns métodos gerais de modificação do DOS, abrindo caminho para aqueles que realmente queiram aprofundar-se no assunto.

Não devemos esquecer de que estamos mexendo com um programa extremamente complexo que ocupa aproximadamente 10K em linguagem de máquina. Portanto, sempre devemos ter extremo cuidado ao fazer experiências, particularmente quando se está trabalhando a partir do monitor. Erros do tipo: ter a intenção de digitar A56EG (que aciona diretamente o comando de catálogo) e sem querer escrever A563G (que entra na rotina de inicialização de discos) são muito mais freqüentes do que se possa imaginar.

Como já foi dito nos artigos anteriores, todo e qualquer comentário ou crítica dos leitores será de grande valor para a nossa equipe.

```
 0  REM ** Tradutor de Comandos **

         @1983 Por Dentro do Apple

=>100 P = 43140:NP = 28
=>110  DIM A$(NP):NB = 0:J = P - 1:
        HOME : PRINT  TAB( 4);"TRAD
       UTOR DE PALAVRAS RESERVADAS"
       : PRINT : PRINT : PRINT "AGU
       ARDE..."
  120  FOR I = 1 TO NP:A$(I) = ""
  130 J = J + 1:A$(I) = A$(I) +  CHR$
       ( PEEK (J)): IF  PEEK (J) <
       128 THEN 130
  140  NEXT I
  150  PRINT : PRINT : PRINT "SOBRA
       M ";NB;" BYTES.": PRINT "DIG
       ITE O COMANDO A ALTERAR --":
        PRINT " <RETURN> PARA TERMI
       NAR:": INPUT "";A$: IF A$ =
       "" THEN 230
  160  IF  LEN (A$) = 1 THEN A$ =  CHR$
       ( ASC (A$) + 128): GOTO 180
  170 A$ =  LEFT$ (A$, LEN (A$) - 1
       ) +  CHR$ ( ASC ( RIGHT$ (A$
       ,1)) + 128)
  180  FOR I = 1 TO NP: IF A$(I) =
       A$ THEN 200
  190  NEXT I: PRINT : PRINT "ESSA
       PALAVRA NAO EXISTE"; CHR$ (7
       ): GOTO 150
  200 NL =  LEN (A$(I)) + NB: PRINT
       : PRINT "SUBSTITUI'-LO POR (
       ATE' ";NL;" CARACTERES):": INPUT
       "";B$:L =  LEN (B$): IF L >
       NL OR L = 0 THEN  PRINT  CHR$
       (7);: GOTO 200
  210 NB = NB +  LEN (A$(I)) - L: IF
       L = 1 THEN A$(I) =  CHR$ ( ASC
       (B$) + 128): GOTO 150
  220 A$(I) =  LEFT$ (B$,L - 1) +  CHR$
       ( ASC ( RIGHT$ (B$,1)) + 128
       ): GOTO 150
  230  INPUT "POSSO FAZER A ALTERAC
       AO FINAL  (S/N)? ";A$: IF  LEFT$
       (A$,1) = "N" THEN  HOME : PRINT
       "NENHUMA ALTERACAO FOI FEITA
        - TERMINADO.": END
=>240  removida
  250  PRINT : PRINT "AGUARDE...": FOR
       I = 1 TO NP: FOR J = 1 TO  LEN
       (A$(I)): POKE P, ASC ( MID$
       (A$(I),J,1)):P = P + 1: NEXT
       J,I
=>260  HOME : PRINT "FIM DA ALTERAC
       AO.": END
=>270  removida
```

Two-liners

O *two-liner* que nos foi enviado pelo estudante Jacyr V. de Quadros Jr. faz uma apresentação muito interessante de formas geométricas aleatórias na tela. De acordo com ele, como as cores de fundo e do desenho são escolhidas ao acaso, pode acontecer de uma coincidir com a outra; neste caso, o Jacyr aconselha a pressionar ctr1-C, que fará com que o programa volte imediatamente ao seu início evitando uma longa espera com a tela vazia. O programa só pára com RESET.

Vale a pena digitar e rodar o programa CURVAS. Uma das possíveis saídas está reproduzida no *dump* abaixo. Continuem mandando suas contribuições para a nossa seção.

```
0  REM     *** CURVAS ***
        Jacyr V. de Quadros Jr.
           Barueri - SP

5  FOR N = 0 TO 1000: NEXT : HGR
    :Q =  RND (1) * 360: HOME : VTAB
    22: PRINT "CURVA = ";Q: HCOLOR=
     INT ( RND (1) * 2) * 3: HPLOT
    0,0: CALL 62454: HCOLOR=  INT
    ( RND (1) * 6): HPLOT 138,79
    :T = 0:PI =  ATN (1) * 4: ONERR
     GOTO 5
10 K = K + Q:T = T + 1:X = 138 +
    T *  SIN (K / 180 * PI):Y =
    95 - T *  COS (K / 180 * PI)
    : HPLOT  TO X,Y:P = (Q < 40 OR
    Q > 320): HPLOT  TO X + (138
     - X) * P,Y + (95 - Y) * P: HPLOT
     TO X,Y: GOTO 10
```

TK 2000: um compatível com o Apple ou uma nova máquina?

Álvaro A. L. Domingues

Quando o TK 2000 saiu, todos que o viram tentaram classificá-lo em uma das famílias já existentes: a do TK 83 (TK-82C, TK 85, CP 200, Ringo, etc), a do Apple (Maxxi, Spectrum), a do TRS 80 (D 8000, CP 500, Jr da Sysdata, etc.) ou outras (Atari, Sinclair Spectrum, etc.).

Isto parece muito natural, uma vez que, sempre que é lançado um produto nacional na área de informática, ele pode ser incluído numa das famílias já existentes. Entretanto, isto não ocorre com o TK-2000.

Uma nova máquina

O hábito de classificar os computadores nacionais em famílias faz com que, — dizendo, por exemplo, que um produto é compatível com o Apple — se crie uma expectativa de que o produto seja capaz de emular o Apple tanto em hardware quanto em software, através de modificações muito pequenas, e faça tudo o que este computador faz.

É importante ressaltar que o TK-2000, uma nova máquina no mercado, não pode e não deve ser incluída em nenhuma das famílias tradicionais dos computadores. Isto é devido à filosofia de trabalho da Microdigital, que, visando uma melhor relação desempenho/preço, criou um produto com características próprias, que dão ao TK-2000 um melhor desempenho para as necessidades dos usuários nacionais.

Entretanto, a Microdigital optou também por dotá-lo com características do Apple, que o tornam parcialmente compatível com esta máquina.

Semelhanças e diferenças

Todo usuário do Apple sabe que, após comprar a CPU (lógica, eletrônica e teclado), deve fazer alguns investimentos para torná-lo operacional. Inicialmente deve-se providenciar o monitor de vídeo ou então uma placa PAL

QUÃO COMPATÍVEL?

Fita

O TK-2000 lê qualquer fita no formato Applesoft. Grava nos formatos Applesoft e TK-2000. As fitas gravadas no formato Applesoft podem ser lidas por um Apple ou compatível. As gravadas no formato TK-2000 só podem ser lidas por um TK-2000. *Observação:* o fato do TK-2000 poder ler um programa do Apple e vice-versa, não significa que ele poderá ser executado imediatamente, sem modificações.

Programas em BASIC

Alguns programas em BASIC que rodam no Apple podem rodar no TK-2000 sem nenhuma modificação. Outros necessitam algum conhecimento do mapeamento de memória do TK-2000 e outros necessitam de traduções de instruções que constam no Apple e não no TK-2000. Um cuidado especial deve ser tomado em relação às linhas que contém funções PEEK e POKE.

Programas em Linguagem de máquina

Alguns programas rodam diretamente, outros necessitam de modificações pequenas, mas outros necessitam de profundas modificações. Seu usuário deverá ter um bom conhecimento da máquina, a nível de hardware, para poder fazer todas as modificações necessárias.

Disquete

O TK-2000 usa um sistema operacional, o TKDOS 3.3, muito semelhante ao DOS 3.3 do Apple. Os comandos são idênticos, permitindo ao usuário do TK-2000 manipular programas e arquivos em disquetes do Apple. O TK-2000 pode ler e gravar dados e programas num disquete do Apple e vice-versa.
Observação: A execução de programas do Apple no TK e vice-versa não está garantida sem adaptações, apesar de serem lidos a partir do disquete do outro computador.

que transforma o sinal de vídeo de computador em um sinal de televisão colorida (dentro do nosso padrão de transmissão), que pode ser aplicado diretamente à entrada da TV.

Se o usuário estiver interessado em armazenar seus dados e programas, deverá optar ou por fita, comprando um gravador de boa qualidade, ou pelo disquete, melhor, mas que implica num dispêndio extra de capital, talvez até superando o preço da própria CPU!

Uma filosofia de trabalho

Desde o lançamento de seus primeiros produtos, a Microdigital preocupou-se em apresentar produtos com as seguintes características:

a) preço acessível

b) boa qualidade

c) o usuário deveria dispender um mínimo para tornar o computador operacional.

Assim, a linha TK tem apresentado computadores que, para tornarem-se úteis, precisam apenas de um gravador cassete e de um televisor preto e branco comuns.

Isso também ocorre com o TK-2000. Diferentemente do Apple e seus compatíveis, o TK-2000 incorpora as interfaces necessárias para torná-lo útil:

a) interface para TV colorida, no sistema PAL.

b) Controle para dois gravadores cassetes.

c) interface para impressora.

Além disso, o TK-2000 apresenta um canal de som que, ao contrário do Apple, não está localizado no gabinete do computar, mas é transmitido, via cabo de RF para a TV, junto com o sinal de vídeo. Isso permite que seu usuário controle o volume do som emitido pelo computador, obtendo uma qualidade melhor.

Outra coisa que o diferencia do Apple é sua memória. O TK-2000 apresenta uma memória RAM de 64 kB, sendo 48 k acessados diretamente pelo BASIC e o restante acessados por software especial.

Recursos adicionais

O TK-2000 apresenta uma série de recursos que permitem ao seu usuário uma grande flexibilidade de programação. Entre estes recursos, encontramos incorporados à sua ROM, dois programas fundamentais para programação em linguagem de máquina: o Monitor Dissassembler e um Mini-Assembler.

O Monitor-Dissassembler permite ao usuário o TK-2000 decodificar um programa em linguagem de máquina, obtendo seus mneumônicos (códigos-lembretes da linguagem Assembly). Já o miniassembler é um programa que faz a operação inversa: a partir dos mneumônicos da linguagem Assembly, cria códigos em linguagem de máquina.

Além disso, o TK-2000 apresenta 50 símbolos gráficos especiais, comandos em uma tela única (através do modo Control-shift), duas páginas de vídeo, três formas de apresentação de imagem (texto, baixa resolução e alta resolução), seis cores, etc. .

Merece especial atenção o comando Sound, que permite ao usuário, por meio do BASIC, comandar o som de seu computador, ao contrário do Apple, que necessita de comandos em linguagem de máquina bastante difíceis de serem executados pelo usuário comum de microcomputadores.

O Software para o TK-2000

A maioria dos programas em BASIC para o Apple, se forem digitados no TK-2000, podem ser rodados sem alterações. Podemos então dizer que o TK-2000 é compatível com o Applesoft, podendo, inclusive, ler fitas gravadas por um Apple ou compatível.

A leitura e gravação das fitas é feita em dois formatos diferentes: Applesoft e TK-2000.

No formato Applesoft, as fitas são gravadas com a mesma formatação do Apple, podendo ser lidas por qualquer computador desta família.

No formato TK-2000, as fitas só poderão ser lidas por um TK-2000. A razão para este formato ter sido adotado pela Microdigital foi sua facilidade de gravação e reprodução, permitindo inclusive ao usuário monitorar a entrada de programas no computador.

Como o formato Applesoft é compatível com o Apple, os programas desenvolvidos em BASIC do Apple podem ser quase todos rodados no TK-2000. Além disso, a Multisoft está colocando à venda dois aplicativos exclusivos para o TK-2000: o cadastro de clientes e o controle de qualidade.

Na área de jogos já estão disponíveis quatro jogos coloridos, com alta qualidade visual e efeitos sonoros.

Além disso, está previsto para breve o lançamento de uma série de programas, uma linha de jogos e programas comerciais, com uma excelente qualidade gráfica, uma grande variedade de cores e alta resolução. ○

CAPITÃO MACHISTA

Inicialmente, parabenizamos a Celso Bressau por ter obtido, tal qual o Gamal Saty, uma solução parcial do Quebra-Cabeça "LGM ou Mensagem Vega?". Infelizmente ela nos chegou quando já havíamos "fechado" a sétima edição da Microhobby. Quanto à interpretação da mensagem, continuamos aguardando colaborações.

Do Quebra-Cabeça da sexta edição, recebemos a tempo para publicarmos nesta edição quatro soluções que forneceram resposta correta (989), das quais três foram escolhidas: a de Milton de Oliveira, de Zilda G. Plassa e a de Roberto Marques Bekman. Isso porque os programas são muito semelhantes (ocupam cerca de 500 bytes e rodam em aproximadamente um minuto). O quarto programa, de Flávio A. R. Steffen, apesar de correto, demora mais de três horas para apresentar a solução certa.

Prezados Senhores,

Sirvo-me da presente para saudar os amigos da Microhobby e também para apresentar a solução do quebra-cabeça: O Capitão Machista.

A resposta que encontrei foi **989** e o programa, rodando num TK 85 (48 k), segue anexo.

Milton de Oliveira

```
  1 REM O CAPITAO MACHISTA
  2 REM ROBERTO M. BEKMAN
  5 FAST
 10 LET P=0
 20 LET Y=1
 30 LET A$="HMHMHMHMHMHM"
 40 LET P=P+1
 50 LET Y=Y+P
 60 LET X=Y/LEN A$
 70 LET Y=Y-LEN A$*INT X
 80 IF NOT Y THEN LET Y=LEN A$
 90 IF A$(Y)="M" THEN GOTO 20
100 LET A$=A$( TO Y-1)+A$(Y+1 T
O )
110 IF LEN A$>6 THEN GOTO 50
120 PRINT "O CAPITAO ESCOLHEU O
NUMERO ";P+1
```

Prezados amigos da Microhobby,

Envio a resolução do quebra-cabeça da revista número 6, "O Capitão Machista".

Linhas importantes para a interpretação do programa:

— **linha 20**: a circunferência aberta, começando com um homem;

— **linha 50**: determina o número que o capitão contará, como se não houvesse dado nenhuma volta na circunferência;

— **linha 70**: ordena novamente a circunferência aberta.

O número que acharão, após rodar o programa abaixo é **989.**

Zilda G. Plassa

```
  1 REM O CAPITAO MACHISTA
  3 REM ZILDA G. PLASSA
  5 FAST
 10 LET X=0
 20 LET A$="HMHMHMHMHMHM"
 30 LET X=X+1
 40 FOR N=1 TO 6
 50 LET K=X-INT (X/(13-N))*(13-
N)
 55 IF K=0 THEN LET K=13-N
 60 IF A$(K)="M" THEN GOTO 20
 70 LET A$=(A$(K+1 TO LEN A$)+A
$(1 TO K-1) AND K<>13-N)+(A$(1 T
O LEN A$-1) AND K=13-N)
 80 NEXT N
 85 SLOW
 90 PRINT AT 10,4;"O MENOR NUME
RO E:";X
```

RESPOSTA DO QUEBRA-CABEÇA

Prezados Senhores,

Como o Capitão não era um passageiro, além disso se ele fosse lançado ao mar ninguém saberia comandar o barco. A disposição dos passageiros, que acho a mais lógica é a seguinte:

```
            H
        M       M
     H             H
  M        C          M
     H             H
        M       M
            H
```

O capitão no centro ficaria com a tarefa de repetir seis vezes um número por ele escolhido e atirar n'água aquele em que o número caísse.

Espertamente o capitão fez um curto programa em seu TK, ocupando pouco mais de 500 bytes de RAM, indicando o menor número que deveria escolher para nenhuma mulher ser lançada aos tubarões, pois o capitão jamais se rebaixaria a tal ponto. Devido aos insistentes pedidos de um engenheiro a contagem começaria por ele.

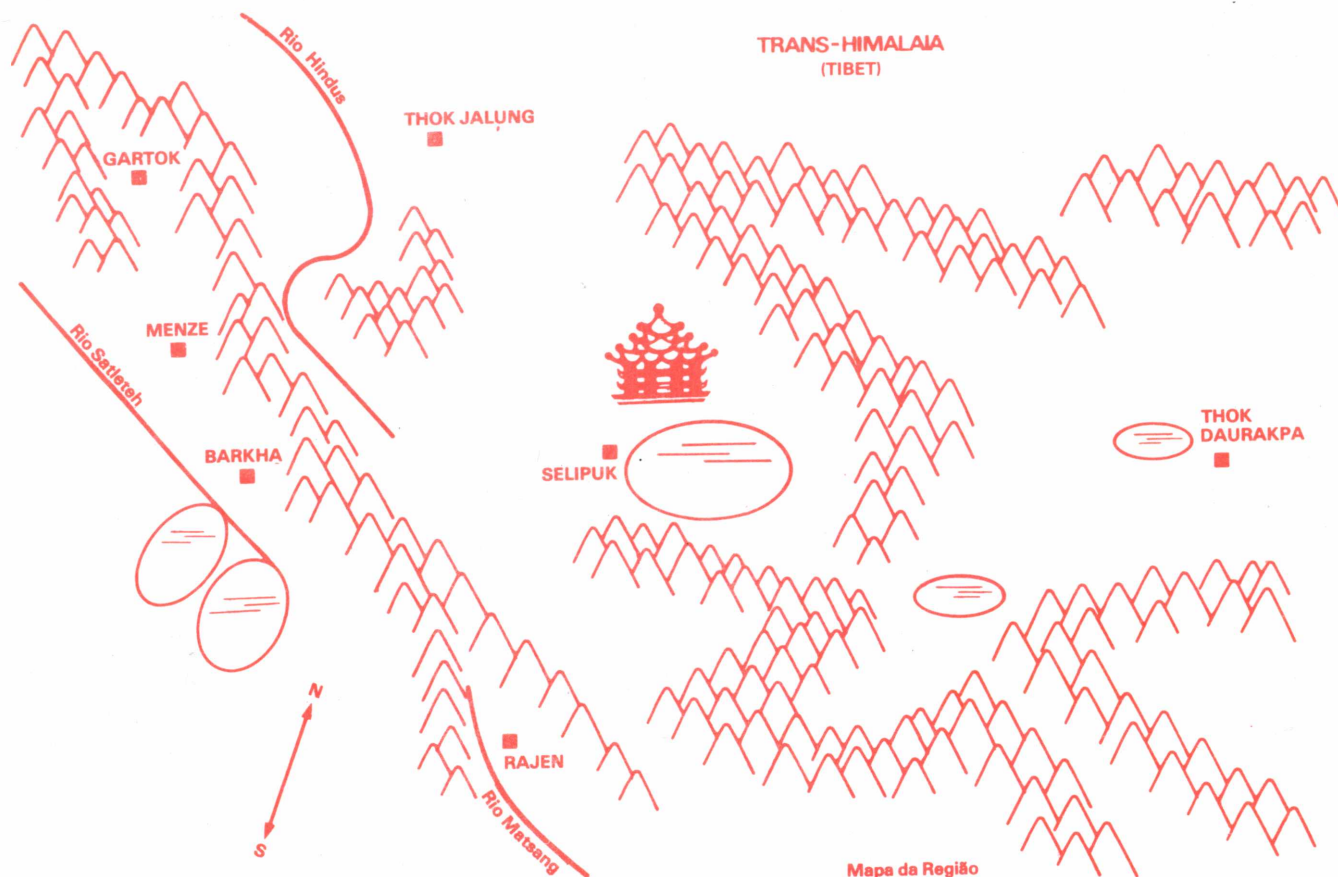
Logo a seguir está listado o meu programa, onde a variável **A$**, acompanhada de devida programação representa a circunferência formada pelos passageiros que a cada rodada tornava-se menor.

Rodando o programa, você verificará que o capitão escolheu o número **989!**

Roberto Marques Bekman

```
  1 REM O CAPITAO MACHISTA
  2 REM MILTON DE OLIVEIRA
  5 FAST
 10 LET A$="HMHMHMHMHMHM"
 20 LET L=0
 30 LET N=3
 40 LET V=N
 50 IF V<=LEN A$ THEN GOTO 80
 60 LET V=N-LEN A$*INT (N/LEN A
$)
 70 IF V=0 THEN LET V=LEN A$
 80 IF A$(V)="H" THEN GOTO 100
 90 IF A$(V)="M" THEN GOTO 140
100 LET A$=A$(V+1 TO )+A$( TO V
-1)
110 LET L=L+1
120 IF L=6 THEN GOTO 180
130 GOTO 40
140 LET L=0
150 LET A$="HMHMHMHMHMHM"
160 LET N=N+2
170 GOTO 40
180 PRINT AT 8,0;"O NUMERO ESCO
LHIDO PELO CAPITAO";AT 10,0;"FOI
:";N
190 STOP
200 SAVE "O CAPITAO MACHISTA"
210 GOTO 5
```

A PEREGRINAÇÃO DO DALAI LAMA

Renato da Silva Oliveira

". . . Tolen enviou um mensageiro à Ajshrangivad para nos chamar à Selipuk. Estamos partindo para Nanda Devi e de lá, seguiremos para Barkha — onde estamos sendo aguardados.

. . .

Eu e Ramarujan já começamos a sentir nosso sangue circular com dificuldade. Nós já percorremos esse caminho muitas vezes no passado, mas nunca o frio e a neve foram tão intensos! Há mais de um dia neva ininterruptamente e a temperatura tem permanecido sempre abaixo de −20ºC.

Apesar de tudo, cremos que com mais um dia de viagem, chegaremos à Barkha.

. . .

Chegamos pela manhã. Somos hóspedes de Shwan Bhugahav, fiel amigo de Tolen. Amanhã seguiremos para Selipuk antes do alvorecer. Desta vez, iremos de avião.

Tolen estará à nossa espera no campo de pouso, onde deveremos chegar pór volta das nove horas.

Até agora tudo tem nos parecido muito estranho. Desde a mensagem que eu e Ramarujan recebemos até a camuflagem do avião — que é branco. Tolen quis que viéssemos clandestinamente para a China e, além disso, pediu-nos para virmos para Barkha a pé, através do Himalaia. Já tentamos obter mais informações com Shwan e parece-nos que ele sabe menos do que nós.

. . .

Chegamos a Selipuk. Tolen levou-nos imediatamente para um ***mosteiro.*** Agora sabemos o motivo de nossa vinda.

Tolen Tonken foi, durante muitos anos, lama fiel ao ***Dalai.*** Quando, em 1959, o ***Dalai Lama*** teve que deixar o Tibet, refugiando-se na India, o ***Panchen Lama,*** mesmo a contragosto, assumiu o controle do Lamaísmo na China. Nesse ano, Tolen isolou-se em Selipuk e acabou deixando de vez a vida nos mosteiros. O ***Dalai Lama,*** entretanto, continua a ter sua fidelidade e a de muitos lamas que permaneceram no Tibet.

A poucas semanas, o ***Dalai*** retornou da India do mesmo modo que nós, e encontra-se aqui em Selipuk.

Por algum misterioso motivo que ele insiste em não revelar, disse-nos que deverá realizar uma peregrinação através de Selipuk, Barkha, Gartok, Rajen, Menze, Thok Jalung e Thok Daurakpa.

Tolen chamou-nos para participarmos da peregrinação com eles, pois conhecemos bem a região e deveremos levar o ***Dalai Lama*** de volta à India quando a terminarmos.

Uma de minhas primeiras preocupações foi encontrar o menor percurso possível, através dos sete vilarejos. Comentei isso com os outros e Tolen trouxe-nos uma tabela, feita por ele, onde encontravam-se as distâncias entre várias localidades da região. Antes mesmo de olharmos a tabela, o Dalai disse-nos qual a seqüência dos locais que deveríamos seguir. Eu, Tolen e Ramarujan nos surpreendemos.

Combinamos partir de Selipuk na manhã seguinte.

À noite, eu não pude resistir a curiosidade de saber se o caminho escolhido pelo ***Dalai Lama*** era realmente o mais curto, dentre os 720 possíveis. Ramarujan e Tolen também queriam saber se seguiríamos pelo menor caminho. Fizemos, então, um rápido programa para meu TK-83 ocupando menos de 2K Bytes de RAM. A partir das distâncias entre cada par das sete localidades, o programa forneceu-nos as duas seqüências mais curtas. Como Tolen já previra e eu e Ramarujan duvidávamos, o ***Dalai Lama*** estava certo!

Amanhã pela manhã, partiremos de Selipuk, e deveremos retornar dentro de pouco menos de um mês.

O percurso escolhido pelo ***Dalai*** entre os dois mais curtos tem, ainda, a vantagem de que o retorno para Selipuk terá menos acidentes geográficos que o início da viagem. Isso parece-me importante, uma vez que estaremos fisicamente mais desgastados . . .''

Estes textos foram extraídos do diário de Nabor Rosenthal. Ele enviou-nos cópias de algumas páginas para não ter que nos escrever sobre tudo o que tem lhe acontecido. Como de hábito, inspiramo-nos nestes acontecimentos para propor o quebra-cabeça desta edição.

Ele se divide em duas partes:

1º) Descobrir, com a ajuda do TK, quais os dois caminhos mais curtos a serem seguidos na peregrinação.

2º) Achar, entre os dois, aquele que foi escolhido pelo ***Dalai.***

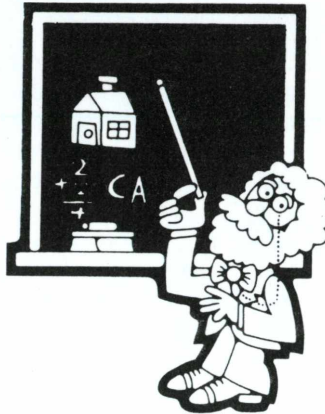
Na abertura de nosso artigo você poderá ver o mapa da região. As distâncias entre cada região estão mostradas na tabela 1.

	SELIPUK	BARKHA	MENZE	RAJEN	GARTOK	THOK JALUNG	THOK DAURAKPA
SELIPUK	—	135	165	120	210	120	218
BARKHA	135	—	60	128	128	135	353
MENZE	165	60	—	180	75	128	390
RAJEN	120	128	180	—	240	210	285
GARTOK	210	128	75	240	—	120	420
THOK JALUNG	120	135	128	210	120	—	300
THOK DAURAKPA	218	353	390	285	420	300	—

Tabela 1: distâncias em Km.

CURSO DE BASIC TK

aula 9

Continuando as STRINGs:

Piazzi — Rossini

Analogamente às funções CODE e CHR$, VAL e STR$ são funções inversas sendo que a primeira fornece uma variável numérica e a segunda uma STRING. (Repare que a notação utilizada coloca um $ no final do nome das das funções que fornecem uma STRING). Estas funções são bastante úteis e permitem interligar os "universos" das variáveis numéricas e das STRINGs pois a primeira (VAL) quando possível, transforma uma STRING no seu valor numérico e a segunda (STR$) transforma um valor numérico ou uma expressão matemática numa STRING. Experimente o seguinte programa que esperamos ser suficiente para esclarecer o efeito de VAL (tecla J):

```
 20 LET B$="2+5-7"
 30 PRINT "B$=";B$
 40 PRINT B$(2 TO 4)
 50 PRINT VAL B$
 60 LET L$="23"
 65 LET Y=VAL L$
 70 LET M$="4"
 80 LET N$="2-47"
 90 PRINT "L$=";L$
 95 PRINT "Y=";Y
100 PRINT "M$=";M$
110 PRINT "N$=";N$
120 PRINT L$+M$
130 PRINT Y+VAL M$
140 PRINT M$+N$
150 PRINT VAL (M$+N$)
160 LET X=5
170 LET A$="X+3"
180 PRINT "X=";X
190 PRINT "A$=";A$
200 PRINT VAL A$
210 LET A$="5*RND"
220 PRINT VAL A$
```

Estude cuidadosamente todas as saídas do programa. Repare novamente na diferença de "espaços" de memória ocupados; por exemplo, na linha 60 temos L$ = "23" ocupando dois espaços um para cada caractere e na linha 65 temos Y = VAL L$, ou seja, 23 também mas agora ocupando apenas um espaço de memória. Em outras palavras, L$ é uma matriz de duas posições enquanto que Y é uma variável simples, apesar de ambas produzirem o mesmo efeito na tela. Preste atenção agora na linha 210; se você digitou a tecla T com o cursor em **F** tudo bem e a linha 220 será executada; no entanto, se você digitou as letras R, N e D separadamente o programa acusará um erro na linha 220. Por que?

Outra utilidade do VAL está no cálculo de expressões ou funções matemáticas. Por exemplo:

```
 10 SLOW
 20 PRINT "ENTRE COM UMA FUNCAO
MATEMATICA QUE USE A VARIAVEL X
"
 30 INPUT Z$
 40 PRINT Z$
 50 PRINT
 60 PRINT "ENTRE COM O VALOR DA
VARIAVEL X"
 70 INPUT X
 75 PRINT X
 80 PRINT
 90 PRINT "VALOR CALCULADO"
100 PRINT VAL Z$
```

Experimente entrar com "5*X**2-2*X+7"; e 5 para X; o que você deverá obter?

Finalmente, apresentamos um exemplo mostrando uma utilização de STR$ (tecla Y) para obter o número de algarismos de um determinado número inteiro e obter alguns de seus dígitos separadamente:

```
10 SLOW
20 LET X=12345678
30 PRINT "X=";X
40 LET A$=STR$ X
50 PRINT "A$=";A$
60 PRINT LEN A$
70 PRINT A$(4)
80 PRINT A$(2 TO 6)
90 PRINT A$(1)+A$(3)+A$(5)
```

A variável X ocupa um espaço de memória para armazenar o número 12.345.678: na linha 40 criamos a variável A$ que ocupa 8 espaços de memória para armazenar os caracteres 1, 2, 3, 4, 5, 6, 7 e 8.

Procure imaginar como seria complicado calcular o número de algarismos de dada variável aritmética ou "extrair" individualmente seus algarismos se não tivéssemos essas funções.

Nota: Na aula 6 mostramos uma limitação da impressora do TK; usando STRINGs, ela pode ser eliminada. De fato, experimente:

```
10 LET X=0.00001
15 LET X$=STR$ X
20 LPRINT X$
```

Um jogo simples utilizando STRINGs: Forca

Vamos apresentar agora uma versão bastante simplificada do jogo da forca, que apresenta algumas limitações mas é bastante útil no sentido de utilização de variáveis STRINGs.

```
 10 SLOW
 20 PRINT TAB 2;"QUAL A PALAVRA
?"
 30 INPUT P$
 40 CLS
 50 LET T=LEN P$
 60 FOR I=1 TO T
 70 PRINT AT 6,2*I;"▒"
 80 NEXT I
 90 LET C=0
100 PRINT AT 10,2;"LETRAS=?"
110 INPUT L$
120 PRINT AT 10,2;"--------"
130 FOR I=1 TO T
140 IF P$(I)<>L$ THEN GOTO 170
150 LET C=C+1
160 PRINT AT 6,2*I;L$
170 NEXT I
180 IF C<>T THEN GOTO 100
190 PRINT AT 10,2;"FIM DO JOGO"
```

Você deve introduzir uma palavra (linha 30) e fazer com que outra pessoa tente "advinhar" a mesma. Note entretanto que nessa versão do jogo você nunca perde . . . Estude com cuidado este programa e repare no uso da variável C como contadora para verificar se você já acertou todas as letras; repare também na formatação das linhas 70 e 160. Outro fato interessante é a possibilidade de se comparar STRINGs:

veja a linha 14Ø . . . Neste ponto você poderia se perguntar se, ao comparar duas STRINGs, pode-se falar em maior ou menor além de igual ou diferente. Felizmente, os códigos dos caracteres no TK estão colocados em ordem crescente para as letras seguindo a ordem do alfabeto; ao comparar STRINGs, o computador compara seus códigos. Assim, a letra A é "menor" que a letra B. Além disso, a palavra "BALA" é menor que "BOLA". De posse dessas informações, experimente então implementar o programa de ordenação numérica dado na aula 7 de forma que ele coloque em ordem ***alfabética***, uma lista de palavras fornecidas por você. Perceba como estas funções que lidam com STRINGs facilitam enormemente o uso de arquivos de dados.

A Função INKEY$

O TK possui uma função bastante interessante chamada INKEY$ e que é útil para a realização de jogos animados; seu efeito é similar ao INPUT, ou seja, ela aceita entrada de dados através do teclado; entretanto, ela só aceita STRINGs formadas por apenas um caractere. Em outras palavras, ela fornece a STRING correspondente à tecla que está sendo pressionada; além disso, ao contrário de INPUT, ela ***não*** interrompe a execução do programa: se no instante em que a instrução que contém INKEY$ está sendo executada, uma tecla é pressionada, ela "entra" no computador; caso contrário, o programa prossegue. Experimente então o seguinte programa (INKEY$ está na tela B):

```
10 SLOW
20 PRINT "*"
30 PRINT INKEY$;
40 GOTO 20
```

Digite RUN e NEW LINE. Começarão a ser impressos asteriscos; antes de acabar a tela, digite então qualquer tecla . . . experimente inclusive digitar NEW LINE e tente explicar o que ocorre.

Vamos agora "complicar" um pouco mais o exemplo:

```
10 SLOW
20 LET X=31
30 LET Y=21
40 PLOT X,Y
50 IF INKEY$="5" THEN LET X=X-
1
60 IF INKEY$="8" THEN LET X=X+
1
70 IF INKEY$="6" THEN LET Y=Y-
1
80 IF INKEY$="7" THEN LET Y=Y+
1
90 GOTO 40
```

Ao executar o programa, aparecerá um "ponto" no centro da tela; experimente movimentá-lo utilizando as teclas 5, 6, 7 e 8 que correspondem às quatro direções indicadas pelas respectivas "flechas". Em aulas futuras veremos maiores detalhes sobre a função INKEY$. O que acontece se você tenta "sair" da tela?

Exercícios

1) Nesta aula foi sugerido que você fizesse um programa que ordenasse em ordem alfabética, uma lista de palavras; implemente então esse programa para que, uma vez feita a lista, ele possibilite a inserção ou retirada de palavras da mesma.

2) Agora com relação ao programa da Força, implemente-o para que ele limite o "número de tentativas" para que seja possível perder o jogo. Tente também fazer com que ele não aceite letras que já foram digitadas e que não aceite também mais de uma letra por vez. Quem sabe você consegue chegar até a desenhar a forca.

3) Implemente o último programa da aula para que, usando as teclas 3, 4, 9 e Ø você consiga movimentar o ponto nas diagonais. A seguir, faça com que a saída apresente linhas tracejadas em vez de contínuas. O

CURSO DE ASSEMBLY Z80
aula 8

Flavio Rossini

Deslocamento de Blocos

Deslocar blocos significa deslocar de uma só vez os conteúdos de uma grande quantidade de memória. Suponha que você tivesse uma sub-rotina razoavelmente grande (por exemplo de 2ØØ bytes) colocada a partir da memória 3ØØØØ e você quisesse transferí-la para começar na memória 2ØØØØ. Isto poderia ser feito com as instruções vistas até aqui porém, acarretando num trabalho razoavelmente grande; entretanto, ele pode ser amenizado parcialmente utilizando a instrução LDI (Load With Increment) que transfere o conteúdo da memória endereçada por DE e a seguir incrementa os pares DE e HL e decrementa o par BC, sem alterar o acumulador.

O código de LDI é 'EDAØ'; dessa forma o programa de transferência poderia ser escrito assim como na figura 1.

Entretanto, há uma maneira melhor de se usar a instrução LDIR onde R significa "repeat"; esta instrução executa o que a LDI faz, até que o valor do par BC seja zero (aqui podemos ver a utilidade do decremento de BC pela instrução LDI)! Assim, se quiséssemos mover 200 bytes deveríamos fazer como na figura 2.

Existem também as instruções LDD (código 'EDA8') e LDDR (código 'EDB8') que em vez de incrementarem os pares de registros DE e HL, os decrementam. (LDD = Load With Decrement). Nenhuma destas instruções afeta o valor do flag de CARRY.

Aplicação prática: SCROLL e ANTI-SCROLL

Vamos agora apresentar um programa que nos permitirá fazer um SCROLL ao contrário. Em outras palavras, vamos fazer a tela descer em vez de subir. Para que ele funcione, é necessário ter uma expansão de memória .

Apenas uma prévia explicação quanto à tela: como todos sabemos, podemos imprimir até 22 linhas com 32 caracteres cada uma; entretanto, no fim de cada linha o computador introduz um código de NEW LINE; portanto, na ***memória,*** tela ocupa 22 x 33 = 726 bytes; que são armazenados ***seqüencialmente.***

Observe o programa da figura 3 (coloque-o a partir da memória 3ØØØØ)

A primeira instrução carrega 726 em BC, que é o número de caracteres da tela (contando os espaços vazios e os NEW LINE). Nos endereços da RAM 16393 e 16397 está outra variável do programa INTERPRETADOR chamada D-FILE que contém o endereço inicial do conteúdo da tela na memória RAM, ***menos um;*** desta forma, ao carregarmos HL com o conteúdo de D-FILE, quando somamos 726, iremos calcular o endereço do último caractere da tela; este endereço é então colocado no par DE.

Colocaremos agora em HL o endereço do caractere que será o último caractere da tela após fazermos este ANTI-SCROLL; ou seja, o endereço do último caractere da penúltima linha (pois a última será perdida), que seria D-FILE + 21 x 33 = D-FILE + 693 o que é feito carregando novamente HL com D-FILE e somando 693. Note que agora o "terreno" está preparado para a instrução LDDR, pois temos em BC precisamente o número de caracteres que deverão ser deslocados (693), em HL (fonte) o endereço do último caractere da 21ª linha e em DE (destino) o endereço do último caractere da 22ª linha; basta apenas "apagar" a 1ª linha da tela, o que pode ser feito utilizando um PRINT AT Ø, Ø; "32 espaços". Dessa forma, utilizando esta sub-rotina podemos fazer:

```
3000 SLOW
3010 FOR I=1 TO 100
3020 PRINT AT USR 30000,0;"-----
---------------------------"
3030 PRINT I
3040 NEXT I
```

Preste atenção à instrução 3Ø2Ø: qual o valor de BC no final da sub-rotina em linguagem de máquina? Será Ø pois BC era o contador. Assim PRINT AT USR 3ØØØØ,Ø equivale a PRINT AT Ø,Ø além de "chamar" a sub-rotina que está a partir da memória 3ØØØØ.

Façamos então uma outra sub-rotina em linguagem de máquina que produz o efeito de SCROLL normal; para isto devemos carregar o ponteiro de fonte (HL) com (D-FILE + 33) e o ponteiro destino (DE) com D-FILE (figura 4).

Note a maneira como 33 foi somado a HL. Isto porque não existe instrução para somar dados ***diretamente*** a pares de registros (ou seja, não existe a instrução ADD HL,33).

Coloque este programa a partir da memória 31ØØØ (basta fazer memória inicial = 31ØØØ no programa HEXA-MEM).

Agora execute:

```
2990 SLOW
3000 FOR I=1 TO 100
3010 PRINT AT 21,USR 31000;"....
............................"
3020 PRINT AT 21,0;I
3030 NEXT I
```

Você poderá fazer jogos gráficos emocionantes usando estas duas subrotinas: basta usar a imaginação . . .

Exercícios:

1) Faça um programa para fazer um SCROLL apenas da METADE SUPERIOR da tela, deixando o resto inalterado. Coloque-o a partir da memória 30000.

2) Faça um programa para fazer um ANTI-SCROLL apenas da METADE INFERIOR da tela, deixando o resto inalterado. Coloque-o a partir da memória 31000.

3) Faça um programa que "encha" a tela de caracteres e, a seguir, "role" a tela para cima e para baixo sob o comando das teclas 6 e 7. Note que ao rolar a tela para cima, a primeira linha deve "reaparecer" na última e vice-versa para baixo.

FIGURA 1

```
LD    DE,20000   '11204E'   carrega ponteiros de memória
LD    HL,30000   '213075'
LDI              'EDA0'     faz transferência e incrementa ponteiro
LDI              'EDA0'
LDI              'EDA0'
. . . e assim por diante . . . 200 vezes
```

FIGURA 2

```
LD    BC, 200    '01C800'   ; carrega contador
LD    DE,20000   '11204E'   ; carrega ponteiros de memória
LD    HL,30000   '213075'
LDIR             'EDB0'     ; transfere, incrementa ponteiros, decrementa
                              o contador e checa o final do loop (BC = 0)
```

FIGURA 3

```
30000  LD BC, 726       '01D802'  ; carrega BC com o "tamanho" da tela
30003  LD HL,(16396)    '2A0C40'  ; carrega em HL o valor de D-FILE
30006  ADD HL,BC        '09'      ; soma HL com BC para obter o en-
                                    dereço do fim da tela
30007  LD D,H           '54'      ; transfere para DE o endereço do fim
                                    da tela
30008  LD E,L           '5D'
30009  LD BC,693        '01B502'  ; carrega BC com o tamanho de 21
                                    linhas (21 x 23)
30012  LD HL,(16396)    '2A0C40'  ; carrega em HL o valor de D-FILE
30015  ADD HL,BC        '09'      ; soma HL com BC para obter o ende-
                                    reço do fim da penultima linha
30016  LDDR             'EDB8'    ; transfere HL para DE , decrementa
                                    ponteiros e contador e testa fim de
                                    LOOP (BC = 0)
30018  RET              'C9'
```

FIGURA 4

```
30000  LD DE,(16396)    'EDB0C40'  ; carrega DE com D-FILE
30004  LD HL,(16396)    '2A0C40'   ; idem para HL
30007  LD A,L           '7D'
30008  ADD A,33         'C621'     ; soma 33 a HL
30010  LD L,A           '6F'
30011  LD A,H           '7C'
30012  ADC A,0          'CE00'     ; transferência do "vai um" (CARRY)
30014  LD H,A           '67'
30015  LD BC, 693       '01B502'   ; carrega o contador
30018  LDIR             'EDB0'     ; loop de transferência
30020  RET              'C9'
```

PEQUENOS

ANÚNCIOS

VENDO ou ***TROCO*** programas para o TK. Os interessados escrevam para: ***Marcelo Nogueira Magalhães.*** Rua Castro Meireles, nº 240. Maraponga, Fortaleza, CE, CEP 60000.

Se você tem TK-82 e TK-83 com expansão de 16 K e gostaria de salvar ou carregar seus programas em 30 segundos, escrevam para mim. ***Tony Shammo.*** Rua Apeninos, 451. São Paulo, SP, CEP 01533.

VENDO TK-82C com Slow e expansão de 16 K mais uma fita Scotch com os seguintes jogos: Redalerj; Crazy-Kong; Passagem para o Infinito; Forca e outros programas. Tratar com ***Marcos Roberto Guimarães*** — Rua José Pinto Coelho — Tel.: 285.3177 de segunda a sexta das 9 às 15 horas. São Paulo.

VENDO ou ***TROCO*** programas de jogos para TK-82, 83, 85 e CP-200 e similares. ***Rangel Cavalcante Filho*** — SQS 309 Bl "A" 103 — Brasília, DF.

VENDO uma fita cassete com pequeno manual explicativo contendo dez programas de 1 K de memória, todos de minha autoria, dos quais sete são jogos. Podem ser carregados em qualquer equipamento da linha Sinclair com ou sem SLOW. Tratar com ***Marcos Saito*** — Rua Visconde de Pirajá, 463, tel.: (011) 63.3324 — SP.

TROCO programas (jogos e aplicativos) para TK e similares. Os interessados escrevam para ***Pedro José da C. e Silva.*** Rua 223, nº 9. Conforto, Volta Redonda — RJ — CEP 27180.

TROCO programas para TK/NE/Sinclair. Mande uma fita com dois programas e receba-a de volta com mais dois programas à sua escolha. Escreva. ***Carlos Sousa Oliveira.*** Rua Artur Prado, 57 — aptº 113. CEP 01322 — Bela Vista, SP. Fone: 289.3416.

RESPOSTAS DO CURSO DE ASSEMBLY

aula nº 7

1)

```
30000   LD BC,0       01 00 00
30003   LD HL,0       21 00 00
30006   ADD HL, BC    09
30007   LD B,H        44
30008   LD C,L        4D
30009   RET           C9
```

2)

```
5000 INPUT A
5005 INPUT B
5010 POKE 31001,A-256*INT (A/256
)
5015 POKE 31002,INT (A/256)
5020 POKE 31004,B-256*INT (B/256
)
5025 POKE 31005,INT (B/256)
5028 SCROLL
5030 PRINT A;"-";B;"=";
5035 PRINT USR 31000
5038 SCROLL
5040 PRINT
5050 GOTO 5000
```

```
31000   LD BC,0       01 00 00
31003   LD HL,0       21 00 00
31006   SBC HL,BC     ED 42
31008   LD B,H        44
31009   LD C,L        4D
31010   RET           C9
```

3)

```
20000   LD DE, 20063  11 5F 4E
20003   LD HL,20071   21 67 4E
20006   LD A,(DE)     1A
20007   LD (HL),A     77
20008   INC DE        13
20009   INC HL        23
20010   LD A,(DE)     1A
20011   LD (HL),A     77
20012   INC DE        13
20013   INC HL        23
20014   LD A,(DE)     1A
20015   LD (HL),A     77
20016   INC DE        13
20017   INC HL        23
20018   LD A,(DE)     1A
20019   LD (HL),A     77
20020   LD BC.20067   01 63 4E
20023   LD HL,20071   21 67 4E
20026   LD A, (BC)    0A
20027   ADD A,(HL)    86
20028   LD (HL),A     77
20029   LD A,0        3E 00
20030   ADC A,0       CE 00
20033   INC HL        23
20034   INC BC        03
20035   ADD A,(HL)    86
20036   LD (HL),A     77
20037   LD A,(BC)     0A
20038   ADD A,(HL)    86
20039   LD(HL),A      77
20040   LD A,0        3E 00
20042   ADC A,0       CE 00
20044   INC HL        23
20045   INC BC        03
20046   ADD A,(HL)    86
20047   LD (HL),A     77
20048   LD A,(BC)     0A
20049   ADD A,(HL)    86
20050   LD (HL),A     77
20051   LD A,0        3E 00
20053   ADC A,0       CE 00
20055   INC HL        23
20056   INC BC        03
20057   ADD A,(HL)    86
20058   LD (HL),A     77
20059   LD A,(BC)     0A
20060   ADD A,(HL)    86
20061   LD (HL),A     77
20062   RET           C9
```

Digite os códigos do programa em linguagem de máquina e depois digite este programa BASIC para testá-lo.

```
   1 DIM A(2)
   5 DIM B(4)
  10 FOR I=1 TO 2
  20 INPUT A(I)
  25 LET C=A(I)
  30 LET B(1)=INT (C/16777216)
  35 LET C=C-16777216*B(1)
  40 LET B(2)=INT (C/65536)
  45 LET C=C-65536*B(2)
  50 LET B(3)=INT (C/256)
  55 LET C=C-256*B(3)
  60 LET B(4)=C
  70 FOR F=4 TO 1 STEP -1
  80 POKE ((I-1)*4+5-F+20062),B(
F)
  90 NEXT F
 100 NEXT I
 110 RAND USR 20000
 115 SCROLL
 120 PRINT A(1)
 125 SCROLL
 130 PRINT A(2)
 135 SCROLL
 140 PRINT PEEK 20071+256*PEEK 2
0072+65536*PEEK 20073+16777216*P
EEK 20074
 145 SCROLL
 150 RUN
```

Quem é você ?
O que você espera de nós ?

micromega

Estas perguntas são imprecindíveis para podermos iniciar um diálogo mais sério, uma vez que queremos que a Microhobby seja o seu livro de cabeceira (ou melhor, o livro de cabeceira do seu micro).

Tentamos fazer isso da melhor maneira possível, através das cartas que nos tem chegado, com elogíos, críticas e sugestões. Porém, sentimos que isso não é suficiente e resolvemos fazer uma pesquisa.

A seguir, você encontrará uma série de perguntas que você pode responder anonimamente ou não (o preenchimento do nome é opcional). Além disso, você terá oportunidade de criticar-nos e dar sugestões que nos serão valiosas. É MUITO IMPORTANTE QUE VOCÊ RESPONDA A ESTE QUESTIONÁRIO. Afinal, trata-se de sua revista e queremos ouvi-lo para traçar o nosso rumo futuro!...

Nome: ______ (opcional)

Residência: ______ (opcional)

Qual a cidade e o estado onde você mora? ______

ATIVIDADE PROFISSIONAL:

a) Atividades ligadas a processamento de dados:
- ☐ Digitador
- ☐ Programador
- ☐ Analista de Sistemas
- ☐ Outros (especificar) ______

b) Profissionais liberais:
- ☐ Advogado
- ☐ Dentista
- ☐ Médico
- ☐ Outros (especificar) ______

c) Área Técnica:
- ☐ Técnico
  Área ______
- ☐ Engenheiro
  Área ______
- ☐ Outros (especificar) ______

d) Estudante:
- ☐ I grau
- ☐ II grau
- ☐ Superior
- ☐ Pós-Graduação: ______
  Curso: ______

FAIXA ETÁRIA:
- ☐ menos de 14 anos
- ☐ de 14 a 18 anos
- ☐ de 19 a 25 anos
- ☐ de 25 a 30 anos
- ☐ de 31 a 35 anos
- ☐ maior que 35 anos

COMO MICROHOBBY CHEGA A SUAS MÃOS?
- ☐ Assinatura ______
  (especificar a partir de que número é assinante)
- ☐ Bancas
- ☐ Empréstimo

CONTEÚDO EDITORIAL:
- ☐ permanecer como está
- ☐ abranger outros computadores
- ☐ dedicar-se mais a principiantes
- ☐ maior números de aplicativos
- ☐ maior número de jogos
- ☐ outros (especificar) ______

O QUE VOCÊ SUGERE PARA MELHORAR A MICROHOBBY?

Relação das lojas autorizadas a receber assinaturas

ALAGOAS

EXPOENTE COM. E REPRESENTAÇÕES LTDA. — Av. Siqueira Campos, 838 — Maceió.

JOALHERIA E BOUTIQUE GLORIA LTDA. Pça. da Independência, 07 — Palmeira dos Indios — AL.

BAHIA

OFICCINA MINI E MICROCOMPUTADORES LTDA. — Shopping Center Itaigara — Lj. 40 — 1º Pav. — Salvador.

CEARÁ

MICROCENTER COMPUTADORES LTDA. Av. Santos Dumont, 2749 — Fortaleza.

SISCOMP SISTEMAS E COMPUTADORES LTDA. — Rua Tibúrcio Cavalcante, 298 — Fortaleza.

GOIÁS

CASA DO MICROCOMPUTADOR SIST. E PROC. DE DADOS LTDA. — Av. Anhanguera, 2574 — Goiânia.

NASA SHOP EQUIP. ELETRÔNICOS LTDA. Rua 4, nº 1042 — Goiânia.

MINAS GERAIS

COMPUTRONIX VENDAS E SERVIÇOS LTDA. — Rua Sergipe, 1422 — Belo Horizonte.

DIDADOS INFORMÁTICA E ADM. LTDA. Rua Minas Gerais, 655 — S/602 — Divinópolis.

DIVIDATA PROCESSAMENTO DEW DADOS S/C LTDA. — Rua Rio de Janeiro, 1023 C.P. 158 — Divinópolis — MG.

MAPSS — ENG. COM. E IND. LTDA. — Rua Getúlio Vargas, 186 — Teófilo Otoni.

MICROESPAÇO COM. E REPRESENTAÇÕES LTDA. — Av. Barão do Rio Branco, 2288/1501 — Juiz de Fora.

MICRO POÇOS LTDA. — Rua Prefeito Chagas, 117 — Poços de Caldas — MG.

MICRON INFORMÁTICA LTDA. — Rua Benjamin Constant, 56 — S/804 — Viçosa.

MIKRO INFORMÁTICA LTDA. — Av. Afonso Pena, 952 — S/627 — Belo Horizonte — MG.

SIETEL SERV. INST. ELÉTRICAS E TEL. LTDA. — Rua Coronel Joaquim Neto, 32 — Santa Rita do Sapucaí.

MATO GROSSO DO SUL

D.R.L. ORG. EMPRESARIAL LTDA. — Av. Afonso Pena, 2081 — Lj. 9 — Campo Grande.

PARÁ

COMPUBEL — COMPUTADORES SIST. E SUPRIMENTOS LTDA. — Rua Quintino Bocaiuva, 1779 — Belém.

DISCOL DISTRIBUIÇÃO E COM. LTDA. — Rua 28 de Setembro, 746 — Belém.

PARANÁ

COMPUSTORE — Rua Emiliano Perneta, 509 — Curitiba.

GRUPO D.G.B. CONSULTORIA ADM. EMPRESARIAL S/C LTDA. — Rua Dr. Murici, 706 — Ala B — 1º andar — Curitiba.

MADISON S/A IMPORTAÇÃO E COMÉRCIO — Rua Mal. Deodoro, 311 — Curitiba — PR.

MORGEN — COM. DE COMPUTADORES Rua Mal. Deodoro, 51 — Galeria Ritz, — 14º and. — S/1405-A — Curitiba.

SHOP COMPUTER CEDM LTDA. — Av. São Paulo, 718 — Londrina.

PERNAMBUCO

ELETROSOM LTDA. — Rua da Concórdia, 287 — Recife.

ELÓGICA MICRO SISTEMAS LTDA. — Rua da Hora, 88 — Recife.

NOVA ERA MICROINFORMÁTICA — Rua Moisés Correa e Silva, 60 — Recife.

SOUZA'S COMPUTER CENTER LTDA. — Rua mª Carolina, 205 — Loja 05 — Boa Viagem — Recife — PE.

TELEVÍDEO LTDA. — Rua Marques do Herval, 157 — Recife.

RIO DE JANEIRO

BEL-BAZAR ELETRÔNICO LTDA. — Av. Almirante Barroso, 81 — Lj. C — Rio de Janeiro.

BRASIL TRADE CENTER COM. E PARTICIPAÇÕES S/A — Av. Epitácio Pessoa, 280 — Rio de Janeiro.

CESPRO — CURSOS DE ESPECIALIZAÇÃO PROFISSIONAL LTDA. — rua República Árabe da Síria, 15 — S/207 — Rio de Janeiro.

COMPUTER CENTER MICROCOMPUTADORES MÁQ. E SISTEMAS LTDA. — Rua Lopes Trovão, 134 — Slj. 247 — Center V — Niterói.

CONSISTEM — CONSULTORIA E TEC. EM MICROINFORMÁTICA LTDA. — Rua do Catete, 311 — Sala 318 — Rio de Janeiro — RJ.

ELDORADO COMPUTADORES E SIST. LTDA. — Rua Visc. de Pirajá, 351 — Lj. 213/214 — Rio de Janeiro.

FOTO ÓTICA RETROPOLIS LTDA. — Rua do Imperador, 715 — Petrópolis.

KRISTIAN TELECOMUNICAÇÕES LTDA. Rua da Lapa, 120/505 — Rio de Janeiro.

MASER — MÁQ. E SERV. DE PROCESSAMENTO DE DADOS — Estr. da Cacuia, 231. Ilha do Governador — RJ.

MICROBYTE SISTEMA E EQUIPAMENTOS COMPUTACIONAIS LTDA. — Rua Buenos Aires, 41 — 3º and. — Rio de Janeiro.

MICRO CENTER INFORMÁTICA LTDA. — Rua Conde do Bonfim, 229 — Lj. 310/2 — Rio de Janeiro.

MICRO HOUSE REPR. LTDA. — Rua Visconde de Pirajá, 547 — S/307 — Rio de Janeiro.

PROSERV PROC. DE DADOS CURSOS E REP. LTDA. — Lgo. 9 de Abril, 27 — S/628 — Volta Redonda.

TELEMÁTICA COM. E IND. LTDA. — Rua Figueiredo de Magalhães, 286 — S/611 — Rio de Janeiro.

RIO GRANDE DO NORTE

INTERMIDIA COMPUTADORES E INFORMÁTICA LTDA. — Av. Nascimento de Castro, 1913 — Lagoa Nova Natal — RN.

RIO GRANDE DO SUL

MAURITINO PIRES SILVEIRA — Rua Manduca Rodrigues, 924 — Sant'Ana do Livramento.

METALDATA ENG. E PROC. LTDA. — Rua Álvaro Chaves, 154 — Cj. 302 — Porto Alegre.

MICROCENTER COMPUTADORES LTDA. Rua 1ª de Março, 113 — Ed. Integral — Cj. 202 — São Leopoldo — RS.

MICROMEGA COMP. E SISTEMAS LTDA. Rua Julio de Castilhos , 441 — 1º and. — Novo Hamburgo — RS.

SIERRA REPRESENTAÇÕES — Av. Farrapós, 2287 — Porto Alegre.

SISTEMÁTIKA COMPUTADORES E SISTEMAS — Rua Andrade Neves, 248 — Pelotas.

SANTA CATARINA

COMPUTERVILLE MICROCOMPUTADORES LTDA. — Rua Tijucas, 375 — Joinvile.

ENTEC REPRESENTAÇÕES LTDA. — Rua Lauro Muller, 700 — Itajaí.

MICRODADOS COMP. SERV. LTDA. — Rua Anita Garibaldi, 8 — Slj. 1 e 2 — Florianópolis.

MICRO HOME MICROCOMPUTADORES E SISTEMAS LTDA. — Rua Pedro Soares, 9 — Florianópolis — SC.

SOME — SOC. MERCANTIL E INDL. LTDA. Rua 15 de Novembro, 1139 — Blumenau.

SÃO PAULO

A.D. DATA COM. SERV. DE INFORMÁTICA LTDA. — Rua João Ramalho, 818 — São Paulo.

ACACIA COM. EXPORT. E IMPORTAÇÃO LTDA. — Av. Paulista, 2073 — Cj. 216/7 — São Paulo.

AGÊNCIA AVANT-GARDE — Av. Brig. Faria Lima, 1237 — Lj. 07 — São Paulo.

ALLCOLOR COM. E REPRESENTAÇÕES LTDA. — Rua Carlos Porto, 85 — Jacareí.

BENNY FEIRA PERMANENTE DE MICROCOMPUTADORES LTDA. — Rua Domingos de Moraes, 407 — São Paulo — SP.

CENADIN — CENTRO NAC. DESENV. DA INFORMÁTICA — Rua José Maria Lisboa, 580 — São Paulo.

CHIP SHOP COMPUTADORES LTDA. — Rua Ofélia, 248 — São Paulo.

COMPUTE — COM. SUPRIMENTOS E EQUIP. ELETRÔNICOS LTDA. — Rua Cruz Machado, 474 C.P. 1427 — Curitiba — PR.

COMPUTEC LTDA. — Rua Mairinque, 66 — São Paulo.

COMPUTER HOUSE — JOÃO CANDIDO COLLADO — Av. Andrade Neves, 1254 — Campinas.

COPEC COMPUTADORES, PROGRAMAS E COM. S/A — Rua Prof. Carlos de Carvalho, 164 — 6º and. — São Paulo.

DATA SOLUTION LTDA. — Av. Eusébio Matoso, 654 — São Paulo.

ENSICOM — ENG. DE SISTEMAS E COMP. Rua Marques do Herval, 409 — 1º andar — S/15 — Taubaté.

EXATRON INFORMÁTICA E ELETRÔNICA LTDA. — Al. dos Arapanés, 841 — São Paulo.

GUAÇUMAQ — MÁQ. E EQ. P/ ESCRITÓRIOS LTDA. — Rua Antonio Gonçalves Teixeira, 97 — Mogi Guaçu.

GUARANI PRESENTES — Av. Senador Vergueiro, 4964 — 1º andar — s/6 — São Bernardo do Campo.

HECTOR A. FERNANDEZ — MIRAGE CINE FOTO — Rua Gal. Câmara, 648 — Santa Bárbara D'Oeste.

INFOR-MATIC INFORMÁTICA E AUTOMAÇÃO LTDA. — Av. Açocê, 309 — São Paulo.

LIDADOS SERVIÇOS E COM. DE COMPUTADORES LTDA. — Rua 7 de Setembro, 876 — Limeira.

LIVRARIA POLIEDRO — Rua Aurora, 704 — São Paulo.

LOG COMPUTADORES LTDA. — Pça. Cândido Dias Castejon, 34 — Slj. — São José dos Campos.

MEMOCARDS — MATERIAIS DIDÁTICOS LTDA. — Rua Amador Bueno, 855 — Ribeirão Preto.

MICRODATA IMPLANTAÇÃO FÍSICA SIST. LTDA. — Rua Montreal, 16 — São Paulo.

MICROCOMP COMPUTADORES LTDA. — Av. Pedroso de Moraes, 1234 — São Paulo.

MICRO PROCESS COMPUTADORES LTDA. — Al. Lorena, 1310 — Lj. 3 — São Paulo.

NÚCLEO DE ORIENTAÇÃO DE ESTUDOS Av. Brig. Faria Lima, 1451 — S/31 — São Paulo.

NADAIS EQUIP. DE SOM LTDA. — Rua Amador Bueno, 213 — Santos.

PRO-ELETRÔNICA COML. LTDA. — Rua Santa Ifigênia, 568 — São Paulo.

RC MICROCOMPUTADORES LTDA. — Av. Estados Unidos, 983 — Piracicaba.

RITZ CINE FOTO LTDA. — Rua Frei Caneca, 7 — Santos.

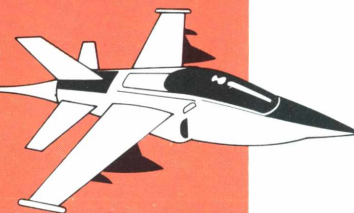
SCHOCK ELETRÔNICA LTDA. — Rua Pe. Luiz, 278 — Sorocaba.

SIPRO COMPUTADORES LTDA. — Av. Carlos Gomes, 396 — Marília.

TELEDALTO ELETRÔNICA E TELECOM. LTDA. — Rua 13 de Maio, 271 — S/101 — Catanduva.

TWIQUI COMPUTADORES E ACESSÓRIOS LTDA. — Av. Paulo Faccini, 72-A — S/6 — Guarulhos.

Extensivo do Anglo: partida em 3 horários para as melhores faculdades.

O Anglo manhã é o curso ideal para você que já terminou o colegial e pode dar o máximo de si no vestibular.

O Anglo tarde é um curso especialmente planejado para quem faz colégio de manhã.

O Anglo noite é um curso que respeita quem trabalha e quer entrar na faculdade.

PRÓXIMAS PARTIDAS: EM MAIO

anglo VESTIBULARES

endereços: R. Tamandaré, 596 - SP.
R. Sergipe, 58 - SP.

MICROMEGA Publicações e Material Didático Ltda.
Caixa Postal 54096 - CEP 01296 - São Paulo - SP
CGC.: 52.275.724/0001-41 - Inscr. Est.: 110.862.362

PEDIDO DE LIVROS E NÚMEROS ATRASADOS

Quantidade	Especificação	até 30/04/84	até 31/07/84
	Linguagem de máquina para o TK	**Cr$ 8.000,00**	9.900,00
	Curso de jogos em Basic TK	**Cr$ 3.400,00**	4.900,00
	Coleção de programas vol. I	**Cr$ 3.800,00**	5.500,00
	Coleção de programas vol. II	**Cr$ 4.000,00**	4.500,00
	Basic TK	**Cr$ 5.000,00**	6.500,00
	Nºs atrasados (nº 2, nº 4, nº 6)	**Cr$ 1 600,00**	
	Fita com São Paulo (1k) e Mansão Maluca	**Cr$ 6.000,00**	
	Fita com Pulga (2k) e Simulador de Vôo (16k)	**Cr$ 6.000,00**	
		Total	

NOME

ENDEREÇO

CEP | CIDADE | EST.

Sim, desejo receber os itens assinalados acima.

____/____/____ Data

______________________ ASSINATURA

Envie seu cheque nominal e cruzado, ou vale postal para Micromega P.M.D. Ltda.
Caixa Postal 54096 - CEP 01296 - São Paulo - SP

☐ em cheque nominal nº ________ Banco ____________

☐ vale postal

Aguarde em média 30 dias para atendimento do pedido.

Não deixe de ler estes livros.

Basic TK

Um livro destinado a quem se interessa em aprender a linguagem do computador TK82-C, 83, 85, CP-200, Ringo, AS-1000 e outros compatíveis com a lógica Sinclair. Complementando os manuais destes computadores, o livro BASIC TK é um auxiliar útil mesmo para os que já possuem alguns conhecimentos sobre sua máquina.

Linguagem de máquina para o TK

Programar em linguagem de máquina nos permite criar programas muito mais rápidos e versáteis que os programados em BASIC. O livro LINGUAGEM DE MÁQUINA PARA O TK ensina, passo a passo e de uma maneira muito leve os segredos desta arte, tornando-o capaz de elaborar jogos e aplicativos nesta modalidade de programação.

Coleção de programas vol. I e II

Programas de todas as modalidades e para todas as idades. É um livro ideal para você que gosta de programas "prontos para uso" para o seu computador.

Curso de Jogos para o TK

Certamente os jogos de vídeo são a coqueluche do momento. Que tal você mesmo bolar seus jogos em seu TK ou compatível? Este livro lhe dá fundamentos para que você possa iniciar-se neste fascinante hobby.

A REVISTA DOS USUÁRIOS DO TK MICROHOBBY

cada vez melhor!

Faça sua assinatura, preenchendo o cupom anexo e ganhe uma tabela com informações utilíssimas para o usuário do TK e compatíveis, tais como mapa da RAM, tabela de conversão decimal para hexadecimal, etc. Além disso você receberá um exemplar da edição especial da MICROHOBBY contendo os melhores programas já publicados.

O que contém uma fita com programas? Programas. Programas aplicativos ou jogos. Entretanto, estas fitas, embora possam proporcionar uma série de funções úteis, fornecem como informação extra apenas uma listagem, que nem sempre é esclarecedora.

Além disso, a Microhobby é uma revista didática, destinada a programadores em vários níveis, desde o principiante até o hobbysta mais ousado, que se aventure a programar em linguagem de máquina.

Receba em sua casa a revista que contém programas, informações, dicas e tudo que você quiser saber sobre microcomputadores e programação.